University of Ljubljana
Faculty of computer and information science

Biserka Cvetkovska

Predicting weather phenomena using discrete Morse theory

Master thesis

second cycle student program
computer and information science

Mentor: prof. dr. Nežka Mramor-Kosta

Ljubljana, 2015

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Biserka Cvetkovska

Napovedovanje vremenskih pojavov z diskretno Morsovo teorijo

Magistrsko delo

magistrski program druge stopnje
računalništvo in informatika

Mentor: prof. dr. Nežka Mramor-Kosta

Ljubljana, 2015

Declaration of Masters Thesis authorship

I, hereby signed Biserka Cvetkovska am the author of the Master Thesis entitled:

Predicting weather phenomena using discrete Morse theory

With my signature, I declare that:

- the submitted Thesis is my own unaided work under the supervision of prof. dr. Nežka Mramor - Kosta,
- all electronic forms of the Masters Thesis, title (Slovenian, English), abstract (Slovenian, English) and keywords (Slovenian, English) are identical to the printed form of the Masters Thesis,
- I agree with the publication of the electronic form of the Masters Thesis in the collection "Dela FRI".

In Ljubljana, 26. August 2015 Author's signature:

First and foremost, I would like to thank my mentor prof. dr. Nežka Mramor-Kosta for guiding me through the writing of this master thesis and helping me with her constant suggestions and advice for improvement. I would also like to thank prof. dr. Franc Solina, prof. dr. Tomaž Dobravec and prof. dr. Gašper Fijavž for their helpful comments and ideas. Thank you dr. Jure Žabkar, Anton Zgonc and everyone at ARSO for providing us with weather data. At the end, I would like to thank my parents, brother and friends for the unconditional love and support.

To my family.

Contents

List of Figures

List of Tables

List of Acronyms

kratica	angleško	slovensko
TDA	Topological Data Analysis	Topološka analiza podatkov
dBZ	Decibels	Decibeli
ARSO	Slovenian Environment Agency	Agencija Republike Slovenije za okolje
ASCII	American Standard Code for Information Interchange	Ameriška standardna koda za izmenjavo informacij

Povzetek

V magistrskem delu obravnavamo dinamiko sprememb intenzitete padavin skozi časovno zaporedje radarskih slik. Cilj naloge je raziskati možnost uporabe dinamike sprememb v slikah kot dodaten parameter pri avtomatičnem napovedovanju ekstremnih vremenskih pojavov.

K problemu pristopimo s topološkimi metodami, z uporabo enega od obetavnih orodij topološke analize podatkov - diskretne Morsove teorije. V delu pokažemo, kako lahko s pomočjo parametrične diskretne Morsove teorije zasledujemo spremembe v zaporedju vremenskih slik in z izgradnjo bifurkacijskih diagramov prikažemo sledi pomembnih značilnosti v slikah.

Razvita metoda je preizkušena na realnih vremenskih radarskih slikah, pridobljenih od Agencije RS za okolje (ARSO). Z namenom, da v čim večji meri eliminiramo vpliv šuma na slikah in sledimo samo značilnostim, ki so pomembne za naš problem, smo razvili in implementirali tudi metodo za reduciranje šuma v podatkih, ki prav tako temelji na diskretni Morsovi teoriji.

Na koncu predstavimo preprost model za klasifikacijo bifurkacijskih diagramov in prikažemo, kako lahko bifurkacijski diagrami skupaj z intenziteto padavin pomagajo pri napovedovanju ekstremnih vremenskih pojavov. Razvite metode so implementirane v programskem jeziku C#, klasifikacijski model pa je zgrajen v Orange-u.

Ključne besede: topologija, topološka analiza podatkov, diskretna morsova teorija, vreme, napovedovanje vremenskih pojavov, bifurkacijski diagrami

Abstract

In this master thesis we concentrate on the dynamics of changes in the precipitation intensity through a time sequence of radar images and inspect how this can be used as a parameter in automatic severe weather prediction.

We approach the problem from a topological standpoint and use one of the promising tools of topological data analysis - discrete Morse theory. We demonstrate how parametric discrete Morse theory can be used in tracking changes in a sequence of weather images and show how we can follow important features in the given sequence by building bifurcation diagrams.

Our method is evaluated on real data represented as a sequence of weather images taken by a weather radar and provided by the Slovenian Environment Agency (ARSO). In order to avoid noise and to obtain only those features from the images that are significant to our research, we implement a method for reducing noise, which is also based on discrete Morse theory.

At the end we present a simple model for classification of the bifurcation diagrams and show how the bifurcation diagrams and the amount of precipitation intensity could influence prediction of weather phenomena.

The methods for data analysis that were developed as part of this thesis are implemented using the programming language C#. The model for classification of bifurcation diagrams is built using Orange.

Keywords: topology, topological data analysis, discrete morse theory, weather, weather prediction, bifurcation diagrams

Chapter 1

Introduction

Weather forecasts and prediction of weather phenomena have a variety of end uses in many areas and are of great importance to both, individuals and organizations. In particular, severe weather forecasting has a huge impact on utility companies, agriculture, forestry, air traffic and marine, but most importantly helps saving lives and properties. However, predicting weather phenomena can be challenging, particularly because weather is a system that changes rapidly and is often dependent on many, sometimes unknown variables. Because of such and similar dynamics in the weather system, weather forecasting models can be very unstable and sometimes even small changes in the weather data can drastically change the weather forecast for a longer period.

Today, weather predictions are done through computer based numerical models of the atmosphere taking many atmospheric factors into account. Nevertheless, the output of the models is still processed with human help thus requiring pattern recognition skills, knowledge of model biases and knowledge of model performance.

In this master thesis, we approach the problem of weather prediction from a topological standpoint, and show how topological data analysis (TDA) can be used in such problems. We base our algorithm on the principles of discrete Morse theory [1, 2]. Discrete Morse theory determines the close relationship

between the topology of some space M, represented as a cell complex and the critical points of a function f, defined on that space M.

We work with a sequence of images taken by a weather radar and process each image separately by building it as a cubical cellular complex with a function f defined on it. In our case, the function f corresponds to the reflectivity returned to the radar receiver after hitting precipitation at each point in the image. Using discrete Morse theory we acquire the critical cells of the function f. The critical cells correspond to important features in the image, so that our main goal is to try and connect those features and track the changes that might appear in time. We manage to achieve this by building bifurcation diagrams.

Since we deal with real data and noise is always present, we implement a simple method for reducing noise. This method is also based on discrete Morse theory and helps us extract those features from the images that are significant to our research.

In this master thesis, we also present a model for classification of the bifurcation diagrams according to the likelihood of severe weather following the time sequence of the corresponding weather images. Additionally, we propose new suitable parameters that can be used in such classifications.

Chapter 2

Topological data analysis

2.1 Introduction

Topological data analysis (TDA) is a rapidly growing field within mathematics and computer science concerning the study of shapes and their properties. Its aim is to extract the topology and the qualitative properties of the space described by some scientific data, usually represented as a finite state of noisy points sampled from unknown space.

Topological data analysis has developed as response to topological impediments emerging from within geometric problems. In the field of computer graphics, researchers encountered problems of connectivity when reconstructing surfaces from point sets. Very often their heuristics resulted in surfaces with holes and tunnels that had to be detected and removed before some other type of geometric processing could be done [3]. It turned out that such problems could be solved thoroughly by using surface reconstruction algorithms from TDA. Apart from geometry, topological data analysis and its algorithms can be also applied in areas like robotics, biology, sensor networks, computer vision, machine learning and others [4], [5]. All these fields use TDA theorems and algorithms to solve different types of problems from a topological point of view. For example, the algorithm described in [6] shows how TDA can be used in machine learning for extracting qualitative features

from a sequence of images, taken by a robot with an on-board camera. The robot uses the extracted features with other sensory data to learn about the environment. Other TDA algorithms are used in the field of image analysis for tracking features in a given sequence of images. Such examples are very similar to the work we do here. Additional researches and related work can be also found in [7]. In [7] weather predictions based on extrapolating the shape and mass of the clouds in radar images and linear regression for predicting future evolutions are presented. The algorithm presented in this paper shows how topological data analysis can be used for tracking the intensity profile of the cloud mass using a parametric active membrane.

2.2 Digital images as cubical cell complexes

As stated earlier, TDA takes into account only the qualitative properties of a space, namely the number of connected components, holes, tunnels or cavities that do not change after continuous deformations [8]. Since TDA algorithms can only deal with spaces and maps that have a finite representation, the data set has to be presented in a form suitable for such analysis. When the data set represents images (as in this master thesis), the most appropriate presentation is a cubical cellular complex.

The essential difference between an image and a set of gray values is the adjacency relations that exist between the gray values in the image. These relations can be simply defined with 2-dimensional structures similar to 2-dimensional matrices or a rectangular raster. Such representations can only be easily comprehendible for human perception, but not sufficient enough in the field of computer image analysis. Many features that are important for the image analysis, such as the connectivity of different regions, the adjacency, the boundaries, etc. are not expressed entirely in a raster image.

As Kovalevsky [9] shows, a cubical cellular complex is the best topological model for a digital image. In order to understand the definition of a cubical cellular complex, suppose that you are given a grid in some Euclidean space.

The grid gives a decomposition of the Euclidean space into closed cubical cells forming the grid. The collection of cubical cells gives the cubical cellular complex. According to Kovalevsky [9],

Definition 1. *A cellular complex C = (E, B, dim) is a set E of abstract elements called cells provided with a function dim, $E \to \mathbb{Z}$ with non negative values called dimension and an antisymmetric, irreflexive, and transitive binary relation $B \subset E \times E$ called the bounding relation (or the face relation).*

A cell of dimension p (if the dimension is important) will be noted by $\alpha^{(p)}$. A complex is a k-dimensional complex if the maximum dimension of all its elements is less or equal to k.

The boundary relation B indicates the ordered pairs between the elements in E. For example, the pair (α', α''), where $dim(\alpha'') > dim(\alpha')$ and element α' is part of the boundary of α'' is an ordered pair. In this case, we say that the element α' is the face of the element α'' and write $\alpha' < \alpha''$. Following this definition, a 2-dimensional square is the face of a cube, a 0-dimensional point (vertex) is a face of a 1-dimensional edge, etc.

Every cell complex can be presented as a geometric object respecting the boundary data. Examples of cellular complexes are simplicial complexes, cubical complexes and more general polyhedral complexes. A cubical cellular complex C can be realized as a union of cubical cells, where each cubical cell is homeomorphic to a product of closed intervals such that

$$C = [a_1, b_1] \times [a_2, b_2] \times ... \times [a_n, b_n]$$

are glued according to the boundary data.

In Figure 2.1 a geometric representation of a 2-dimensional cubical cellular complex is shown.

2.3 Discrete Morse Theory

Discrete Morse theory is a powerful tool in the field of topological data analysis that helps in the extraction of important features in a given set of points

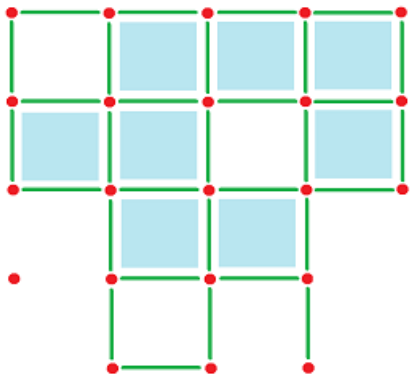

Figure 2.1: An example of a cubical cellular complex. Source [10]

with given values of a function f. It introduces the idea of a critical cell of the function and states that there is a very close relationship between the topology of a given cubical cellular complex C and the critical points of the function f defined on that complex C. The theory was developed by Robert Forman [1], analogous to the smooth Morse theory developed by Morse, 70 years earlier [11]. It is widely used in different fields of mathematics, in data analysis, computer vision, computer science, etc. For example, in graph theory it is used for checking whether a graph is connected and contains cycles, it is used in the field of computer image analysis for extraction of important information from a given image, etc.

In order to be able to find the critical points of a function associated with a given cubical cellular complex C, one must present it as a discrete Morse function over the cellular complex C. A Morse function assigns higher numbers to higher dimensional cells, with at most one exception, locally at each cell. More precisely, a discrete Morse function

$$f : C \to \mathbb{R}$$

assigns a value for every cell $\alpha^{(p)} \in C$ of dimension p ≥ 0, such that

$$\#\{\beta^{(p+1)} > \alpha^{(p)}, f(\beta) \leq f(\alpha)\} \leq 1$$

$$\#\{\gamma^{(p-1)} < \alpha^{(p)}, f(\gamma) \geq f(\alpha)\} \leq 1$$

In Figure 2.2 the function defined on the left is not a discrete Morse function since there exists a 1-dimensional cell with assigned value of 0, which is smaller than its two 0-dimensional faces with assigned values of 2 and 1. The red arrow points out the 1-dimensional cell because of which the given function is not a valid discrete Morse function. On the other hand the function defined on the right is a valid discrete Morse function since all 1-dimensional cells have assigned values bigger than at least one value assigned to one of its faces.

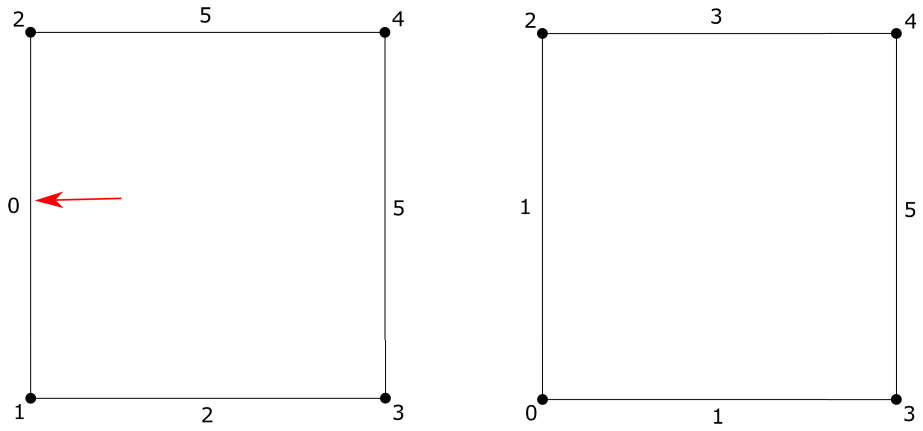

Figure 2.2: The image on the left does not represent a valid discrete Morse function. The image on the right represents a valid discrete Morse function

As stated earlier, another important part of discrete Morse theory is the notion of critical point. A cell $\alpha^{(p)}$ is critical if all its cofaces take strictly greater values and all its faces strictly lower. More precisely,

$$\#\{\beta^{(p+1)} > \alpha^{(p)}, f(\beta) \leq f(\alpha)\} = 0$$

$$\#\{\gamma^{(p-1)} > \alpha^{(p)}, f(\gamma) \geq f(\alpha)\} = 0$$

However, constructing discrete Morse functions and assigning values to each cell in a given cell complex C, can be a rather complex and difficult

process. Therefore, Forman [1] proposes finding the discrete gradient vector field corresponding to a discrete Morse function defined on a cell complex C.

The discrete gradient vector field V, corresponding to the discrete Morse function f is a collection of pairs $(\alpha^{(p)}, \beta^{(p+1)})$, such that:

$$f(\beta) \leq f(\alpha).$$

If we manage to find such a pair, we draw an arrow from α to β. A cellular complex with a discrete Morse function satisfies exactly one of the following:

1. α is the tail of exactly one arrow,
2. α is the head of exactly one arrow,
3. α is not tail nor head of an arrow.

The cells that are not paired with other cells, or more precisely, the cells that are not the tail nor the head of an arrow, are critical cells.

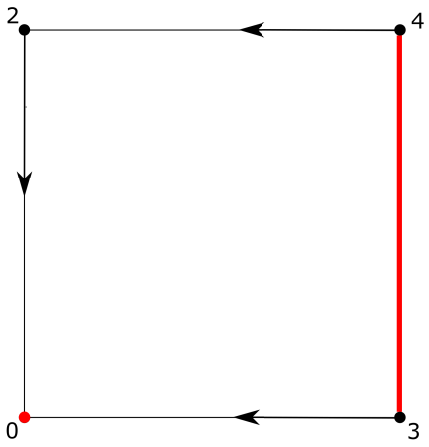

Figure 2.3: The discrete gradient vector field of the Morse function in Figure 2.2

As Forman continues in his "User's guide to discrete Morse theory" [1], it is natural to study the dynamical system induced by flowing along the vector field. Furthermore, he introduces a sequence of simplices along which

the function values decrease. This sequence is called a gradient path. Given a discrete gradient vector field V on a cell complex C, a gradient path is the sequence of cells:

$$\alpha_0^{(p)}, \beta_0^{(p+1)}, \alpha_1^{(p)}, \beta_1^{(p+1)}, \alpha_2^{(p)}, \beta_2^{(p+1)}, ..., \beta_r^{(p+1)}, \alpha_{r+1}^{(p)}$$

such that, for each $i = 0, \ ... \ r$, $\{\alpha < \beta\} \in V$ and $\beta_i > \alpha_{i+1} \neq \alpha_i$. Since function values decrease along gradient paths, a gradient path can not form a cycle. Following these gradient paths we can efficiently follow the dynamics imposed by a function f in a system represented as a cell complex.

Chapter 3

Processing the data

Weather is a system that changes rapidly and is dependent on many parameters. One of these parameters is the precipitation intensity in a scanned volume, which is determined through reflectivity. The reflectivity of a cloud Z, gives the intensity of the returned energy reflected from the cloud. It depends on the number and the amount of rain, snow and hail particles in the cloud and is expressed in decibels (dBZ). dBZ is a logarithmic dimensionless unit which is used in weather radar to estimate the return echoes or reflectivity in a scanned volume [12, 13].

Usually, reflectivity by itself can not distinguish the type of precipitation with certainty. A combination of additional parameters such as dual polarization, Doppler effect, etc. can help meteorologists distinguish among different types of atmospheric phenomena (rain, snow, hail, etc.). In the master thesis we study the dynamics of reflectivity within a sequence of meteorological images as a possible additional parameter for weather predictions. Table 3.1 contains dBZ values that equate to approximate precipitation intensities.

3.1 Data set definition

The data set analyzed in this master thesis is presented with a sequence of radar images taken by a weather radar and obtained from the Slovenian

dBZ	R (mm/h)	Intensity
5	<0.01	Hardly noticeable
10	<0.01	Light mist
15	0.01	Mist
20	0.02	Very light
25	0.05	Light
30	0.1	Light to moderate
35	0.22	Moderate rain
40	0.45	Moderate rain
45	0.92	Moderate to heavy
50	1.90	Heavy
55	4	Very heavy/small hail
55	8	Extreme/moderate hail
65	16.6	Extreme/large hail

Table 3.1: dBZ values that equate to approximate rainfall intensities [12]

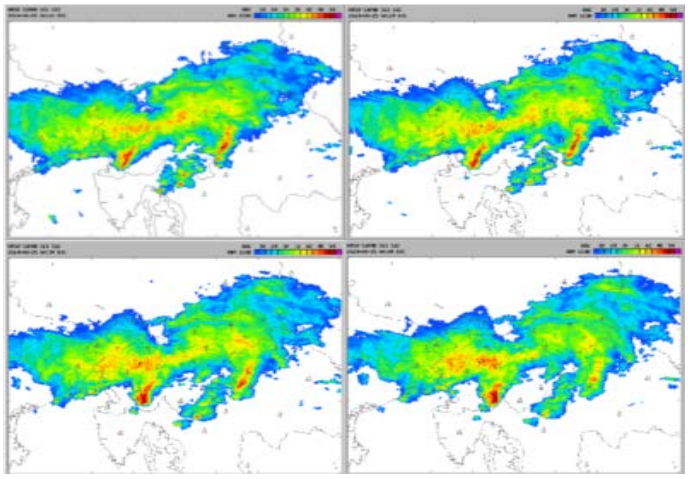

Figure 3.1: Meteorological images taken by a weather radar from ARSO, Slovenia

Environment Agency (ARSO). Each image reflects the meteorological state above Slovenia at a certain period of time. The time gap between two consecutive images in the sequence is 10 minutes. Each image is given in GIF and SRD file format.

In Figure 3.1 examples of meteorological images in GIF format are given. Each image represents a 10 minute time frame and shows the precipitation intensity in that frame.

The SRD format is internal ARSO format which encodes each pixel in the image with an American Standard Code for Information Interchange (ASCII) character and determines the reflectivity. For example the "@" sign equals 12 dBZ. Reflectivity can be computed from an ASCII character by using the formula given below. The parameters used are defined at the beginning of each SRD file.

$$((ASCII - OFFSET) * SLOPE) + START.$$

For example, the "@" character has a value of 64 in the ASCII table, so in the case of an offset of 64, the reflectivity will be equal to the value of START. If the offset is 64, the slope 3 and the start 12, we can conclude that the "@" character represents 12 dBZ. The "A" character that has a value of 65 in the ASCII table represents 15 dBZ, the "B" character represents 18 dBZ, etc. Figure 3.2 shows how certain parts of an image are encoded with SRD.

3.2 Tools used for data processing

The algorithms defined in the next section are written in C#. The solution is presented in a form of a Console Application and is developed using Microsoft Visual Studio 2013 IDE. The second tool used in this master thesis is Orange. Orange [14] is an open source tool used for data visualization and analysis that contains add-ons for data mining, machine learning, bioinformatics and text mining. Here, we use it for classification.

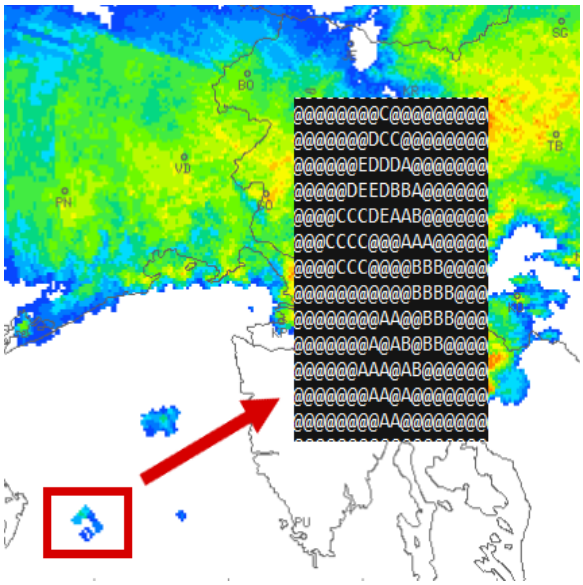

Figure 3.2: Encoding certain parts of the image using SRD encoding

3.3 Algorithms

3.3.1 Constructing cell complexes from the data

In order to use TDA algorithms over the set of data described in Section 3.1, we must first present the data in the form of a cubical cell complex. When constructing the cubical cell complexes, 0-cells are represented by pixels or points. 1-cells are represented by edges that connect two pixels (points), whose coordinates differ between each other by one in a single axis. Additionally, 2-dimensional cells are presented as unit squares.

In Figure 3.3 below, we can see how a cubical complex for a small 5 by 5 2D image is constructed. The second image represents the final result of our algorithm. The red dots represent 0-cells, the green lines represent 1-cells and the blue squares represent 2-cells.

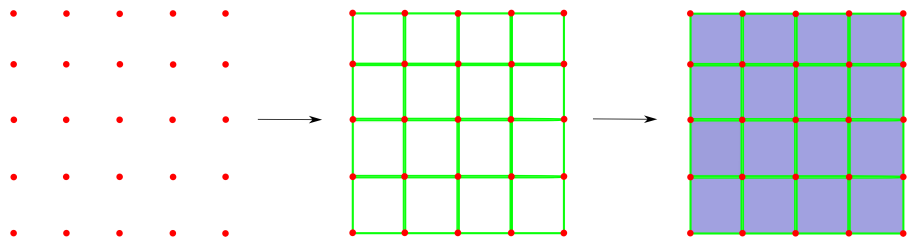

Figure 3.3: Construction of cubical cellular complex for a 2D image. Source [10]

3.3.2 Obtaining the discrete gradient vector field

As explained earlier, our goal is to track changes in a given sequence of meteorological images and follow the precipitation intensities in each image. In order to achieve our goal we need to extract certain parts of the images called critical cells and then see how those parts interact between each other through the given sequence. They can either conjugate into bigger structures or they can decompose into smaller ones. Extracting such information can be done using discrete Morse theory. Similar ideas have been discussed in [15, 16].

A single radar image has a resolution of 401x301 and will correspond to a cubical complex consisting of 120 701 0-dimensional cells, 240 700 1-dimensional cells and 120 000 2-dimensional cells. Once the cubical complex for the image (following the algorithm given in the previous section) is constructed, we need to define a discrete Morse function f on that cubical complex. In our case the function f is the amount of reflectivity which is proportional to the precipitation intensity and is given in the pixels, that are in the vertices of the cubical cellular complex. In order to transform the function into a discrete Morse function we need to determine its discrete gradient vector field. The discrete gradient vector field shows in which direction the function decreases and helps in determining the critical cells of the function. The critical cells of the function f represent its local extremes (minima, maxima) and saddles. Thorough explanation of these terms is given in the

sections below.

3.3.2.1 Process lower stars algorithm

For constructing the gradient vector field, we use the algorithm presented in [15]. It requires that the f value of each 0-cell is distinct, so that the 0-cells in the cubical complex K can be ordered as

$$f(x_0) < f(x_1) < f(x_2) < ... < f(x_N).$$

So, we need to find a way to assure distinct values. In order to achieve this, we perturbe the given function f, such that we add a random double value in the range (0, 1) to each f value

$$f_{pert}(x_0) = f(x_0) + rand(0, 1)$$

$$f_{pert}(x_n) = f(x_n) + rand(0, 1), ...$$

We also define the lower star of each vertex x. The lower star of x is the cubical complex that contains all cofaces of x, that is, all cubical elements that have x as a face for which x is the highest valued cell. We can define the lower star $L(x)$ more precisely in the following form

$$L(x) = \{\alpha \in C \mid x < \alpha \text{ and } f(x) \geq f(\alpha)\}$$

Additionally, we must define a function that computes the number of unpaired faces for a given coface of x. For example, if $L(x)$ is the lower star of a zero dimensional cell x and α_i is a coface of x, the number of unpaired faces of α_i, equals the number of faces of α_i that are contained in $L(x)$ and have not yet been paired or marked as critical.

The algorithm also requires two queues of cells. One which stores cells that have exactly one unpaired face (pqOne) and another which stores cells that have zero unpaired faces (pqZero). These queues require an ordering of the cells. The ordering of the cells in the queues must be determined by the maximum f value of the 0-dimensional cells. This means that higher

dimensional cells (1 and 2-dimensional) must be ranked according to the values of its faces. Any type of ordering that orders the cells according to the rules specified above can be used. In our case we define such ordering by assigning values to higher dimensional cells such that, each 1-cell is assigned with a value that corresponds to the arithmetic mean of the values of its two zero dimensional faces. For example, if α is an edge between two points β and γ with values $f_{pert}(\beta)$ and $f_{pert}(\gamma)$ respectively, the value corresponding to the 1-cell will be equal to the arithmetic mean of $f_{pert}(\beta)$ and $f_{pert}(\gamma)$. Similar, each 2-cell is assigned with a value that represents the arithmetic mean of the values of its four zero dimensional faces.

The algorithm goes through each 0-cell in the cubical complex and examines its lower star. If the lower start of x is empty, then x is critical and is added to the list of critical cells C. Otherwise, x is paired with the front of pqOne, which contains the edge containing the x with the lowest value. The rest of the cells from $L(x)$ which have 0 unpaired faces are added to the pqZero queue and the ones with only one unpaired face are added to pqOne. Bear in mind that both the queues pqOne and pqZero are always ordered using the ordering rule mentioned earlier. For each cell in pqOne, the algorithm tries to create cell pairs by pairing cells from pqOne with their single available non paired face. When pqOne is empty, the pairing is no longer possible and the front cell of pqZero is added to the list of critical points. The pseudo code for the algorithm is given below [15].

Each cell x is a candidate for a cell pair when the number of unpaired faces of x is exactly one, on the other hand, it is a candidate for a critical cell when all its faces have already been paired or marked as critical (the number of unpaired faces is 0). Pairs are added in the list of pairs V and critical cells are added in the list of critical cells C.

The result of this algorithm are two lists. The first one C contains the critical cells and the second one V contains the cell pairs and represents the discrete gradient vector field. Both of these lists are very important and are used in later algorithms.

Algorithm 1 Process Lower Stars

```
1: Input: Cubical Cellular Complex K
2: Output: Discrete Vector Field V
3: Output: Critical Cells C
4: for each k in K do
5:   find LowerStar(k)
6:   sort LowerStar(k)
7:   if LowerStar(k) = 1 and LowerStar(k) = k then
8:     add k to C
9:   else
10:    delta = LowerStar(k).pop()
11:    V[k] = delta
12:    add all other 1-dimensional cells in LowerStar(k) to pqZero
13:    if dim(alpha) >dim(delta) and NumUnpairedFaces(alpha) = 1 then
14:      add all other cells in LowerStar(k) to pqOne
15:    end if
16:    while pqOne >0 or pqZero >0 do
17:      while pqOne >0 do
18:        alpha = pqOne.pop()
19:        if NumUnpairedFaces(alpha) = 0 then
20:          add alpha to pqZero
21:        else
22:          V[pair(alpha)] = alpha
23:          remove pair(alpha) from pqZero
24:          if (dim(beta) >dim(alpha) or dim(beta) >dim(pair(alpha)))
   and NumUnpairedFaces(beta) = 1 then
25:            add all cells beta from LowerStar(k) to pqOne
26:          end if
27:        end if
28:      end while
```

```
29:         if pqZero >0 then
30:           gama = pqZero.pop()
31:           add gama to C
32:           if dim(alpha) >dim(gama) and NumUnpairedFaces(alpha) = 1
    then
33:             add all cells alpha from LowerStar(k) to pqOne
34:           end if
35:         end if
36:       end while
37:     end if
38: end for
```

3.3.2.2 Shapes of critical cells

There are three types of critical cells. A critical cell can be either a maximum, a minumum or a saddle of the discrete Morse function. Zero dimensional cells present local minima, one dimensional cells present saddles and two dimensional cells present maxima. In Figure 3.4 you can see how the values of the function f changes in the direction of the discrete vector field and how these critical cells are created.

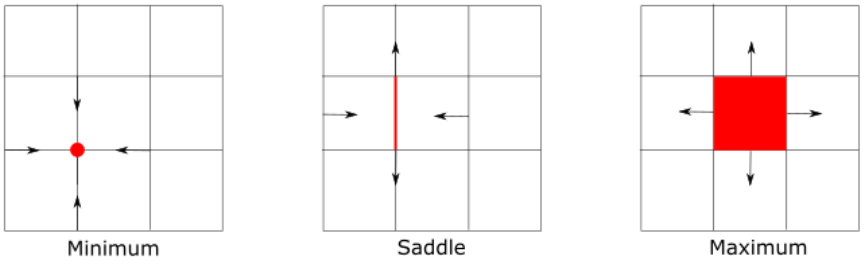

Figure 3.4: Shapes of critical cells

3.3.3 Building the Morse Complex

Finding the Morse chain complex corresponding to a discrete Morse function on a cell complex C can help in computing the homology of C. Furthermore, it can define the exact number of connected components, holes and voids, which form the given complex C. In order to construct the Morse chain complex of C, we need to find the maximal gradient paths of the discrete gradient vector field, which begin in each critical cell $\alpha^{(p)} \in C$ and terminate at another critical (p - 1)-dimensional cell. Knowing the gradient paths and the Morse chain complex is a prerequisite for an algorithm that we need to implement in order to get to the desired results - finding a connection between the given sequence of images. This algorithm will help us in constructing the bifurcations diagrams.

3.3.3.1 Extract Morse complex algorithm

In [15] an algorithm that finds gradient paths between critical points with a simple queue data structure is presented. When building the Morse complex, we followed the implementation of this algorithm, but we also upgraded it so that it can suits our needs. The original version of the algorithm finds which critical cells are connected, but it does not compute the actual gradient paths that lead from a given p-dimenstional critical cell to other (p - 1)-dimensional critical cell. Also, the original algorithm did not take into account the gradient paths between a critical cell and its critical faces. For example, if we have a critical 2-dimensional cell and some of its 1-dimensional faces are also critical, the algorithm does not find such connections. We also succedeed to fix that.

The algorithm goes through each 1 and 2-dimensional critical cell and examines its faces. If a face is critical then there exists a gradient path between the examined critical cell and its face, therefore we add the gradient path to the Facelist list. If the face is not critical then it is worth exploring further, therefore we add the face into the queue data structure for further examination. Once we have added all faces that are not critical in the queue

we continue to explore their faces. A cell is popped from the queue and we check if some of its faces are critical. Non critical cells are added to the queue, critical cells are added to the *Facelist* list. In meantime we keep track of the paths that lead from a given p-dimensional to (p - 1)-dimensional critical cell. When computing the gradient paths we need to have in mind that gradient paths can split in different directions and a single gradient path can be a part of another gradient path. In order to achieve proper detection of gradient paths we also implemented a structure that keeps track of the available gradient paths at all the time for each face we examine. This structure is the *parentPaths* list in the pseudo code given below. *parentPaths* contains the gradient path from a certain critical cell to another regular cell up to the splitting. When the splitting happens, the gradient path can continue in different directions (at most 3), but since we have implemented this structure we will always know how to back track the gradient path that lead to the splitting. The *parenPaths* combined with *paths* list gives us the gradient path from a certain 2-dimensional critical cell to another 1-dimensional critical cell.

Another important part of the algorithm is to clear the list of *paths*, whenever a splitting happens or whenever we reach a 1-dimensional critical cell that needs to be added in the *Facelist* list.

The algorithm returns a *Facelist* list that is in the form of a Dictionary (contains key - value pairs). The key represents the p-dimensional critical cell and the value is an object that represents the (p - 1) dimensional critical cell and the gradient path between the p and the (p - 1)-dimensional cell.

3.3.4 Filtering out inconsequential critical cells

Since we are interested only in parts of the images with high precipitation intensity, we consider critical cells with values below a threshold ϵ as negligible and filter them out. Every 2-dimensional cell which has a value lower than the threshold ϵ is not considered as an important cell, and therefore is filtered out. The value of the 2-dimensional cell is determined by the

Algorithm 2 Build Morse Complex

```
1: Input: Cubical Cellular Complex K
2: Input: Discrete Vector Field V
3: Input: Critical Cells C
4: Output: Facelist with gradient paths
5: for each p in {0, 1, 2} do
6:   for each gama in C do
7:     if  p >0 and dim(gama) = p then
8:       alphaObjects = GetAllFaces(gama)
9:       for each alpha in alphaObjects do
10:        if  alpha is not critical then
11:          add alpha to queue qBfs
12:        else
13:          Facelist[gama] = {alpha, null} {There exists a gradient path between alpha and gama but in this case the path is null since alpha is a face of gama and we can get directly from alpha to gama.}
14:        end if
15:      end for
16:    end if
17:    var paths {Define paths}
18:    var parentPaths {Define parentPaths}
19:    var latestVersionOfPaths {Define latestVersionOfPaths}
20:    while Lenght(qBfs) >0 do
21:      alpha = qBfs.getFirst() {Get the first element}
22:      beta = V[alpha] {Find the cell with which alpha is paired}
23:      paths[alpha] = beta {Keep track of the paths between critical cells}
24:      deltaObjects = GetAllFaces(beta) {Continue inspecting the faces of beta until you reach a critical cell}
25:      latestVersionOfPaths = paths {Keep track of the latest version of paths}
26:      var localPathsTemp {Define localPathsTemp}
27:      if Length(paths) >0 then
28:        starter = paths.getFirst()
29:        if parentPaths.Contains(starter) then
30:          localPathsTemp = parentPaths[starter]
31:          add all gradients paths from paths to localPathsTemp
```

```
32:           else
33:             localPathsTemp = paths
34:           end if
35:         else
36:           localPathsTemp = latestVersionOfPaths
37:         end if
38:         for each delta in deltaObjects do
39:           add all paths from localPathsTemp to parentPaths[delta]
40:         end for
41:         if NumberOfSplittedPaths(deltaObjects) = 0 then
42:           paths = null
43:         end if
44:         for each delta in deltaObjects do
45:           if delta is critical then
46:             Facelist[gama] = {delta, localPathsTemp}
47:             paths = null {Clear all paths that were added so far}
48:           else
49:             qBfs.addFirst(gama) {Add gama at the beginning of the
    queue}
50:           end if
51:         end for
52:       end while
53:     end for
54: end for
```

maximum f value of its 0-dimensional faces. The value of ϵ can be chosen according to the values given in Table 3.1. For example, if we are interested in severe weather phenomena, we set the threshold value to 40 - 42 dBZ. Choosing the threshold ϵ between 40 - 42 dBZ means that everything above these values can be worth examining. We decided not to take into account the rest of the critical cells, which have values lower than the chosen ϵ since these critical cells are inconsequential to our research. Our goal is to look for potential features that can lead to severe weather phenomena, therefore higher f values.

Once we have applied the filters mentioned above, we can start cancelling out critical cells using the cancellation theorem.

3.3.5 Cancellation of critical cells and reducing noise

When working with real data, we usually face the problem of noise. For example, as a result from the algorithm that determines the discrete gradient vector field we obtained many critical points, most of which do not have significant value to us. In order to reduce the number of critical cells and concentrate only on those that are important for our research, we implemented an algorithm for reducing noise. The algorithm is based on the "Cancellation theorem" represented in Forman's paper [1], where he explains how critical cells can be cancelled out in pairs.

Forman's theorem explains

Theorem 1. *If f is a discrete Morse function defined on C such that $\alpha^{(p)}$ and $\beta^{(p+1)}$ are critical cells and there exists exactly one gradient path between the boundary of β and α, then another Morse function g can be defined on C with the same critical cells, such that β and α are no longer critical.*

The algorithm we implemented following Forman's theorem, examines the single gradient path defined between two critical cells β and α. If such path exists, we cancel out both critical cells β and α by reversing the discrete

gradient vector field along the given gradient path. Figure 3.5 shows how the gradient vector field can be reversed using our algorithm.

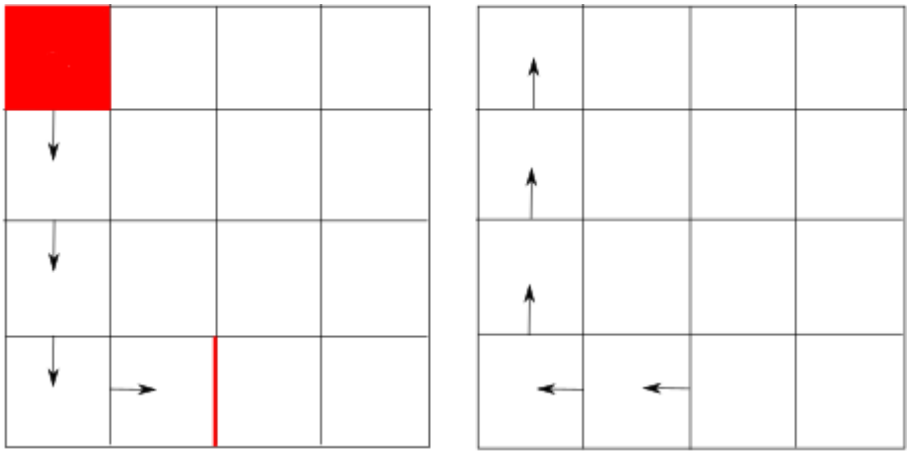

Figure 3.5: Reversing the discrete gradient vector field in order to achieve cancellation of critical points

The algorithm works for both 1-dimensional and 2-dimensional critical cells. Because in this master thesis we are only interested in those critical cells for which the value of the function reaches a local maximum, we only take into account 2-dimensional critical cells.

It is really important to note that we only cancel those 2-dimensional critical cells among which there is only one gradient path and for which the difference among the f values is insignificantly small (shows no indication of change, in our case 1). All other critical cells are left untouched. The reason we cancel out only the 2-dimensional critical cells among which there is only one gradient path is that we can easily create cycles if we start cancelling out the 2-dimensional critical cells among which there is more than one gradient path.

When cancelling out critical cells, we must think about the changes cancelling causes to the other gradient paths in the discrete gradient vector field. Suppose, we cancel out critical cells A and e (see Figure 3.7), there is also another gradient path between a third critical cell B and the cell that is

being cancelled e. In such situations we need to adjust the gradient paths that start from B.

Figures 3.6 and 3.7 explain how the gradient field can be adjusted in such situations. We can see that after cancelling the path from A to e, the path that used to lead from B to e now ends in e' instead of in e.

The final result of the algorithm contains a list of gradient paths that are later used for construction of bifurcation diagrams.

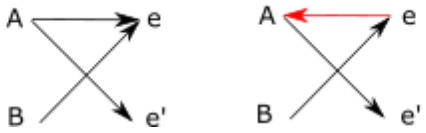

Figure 3.6: Reversing the discrete gradient vector field and adjusting the paths

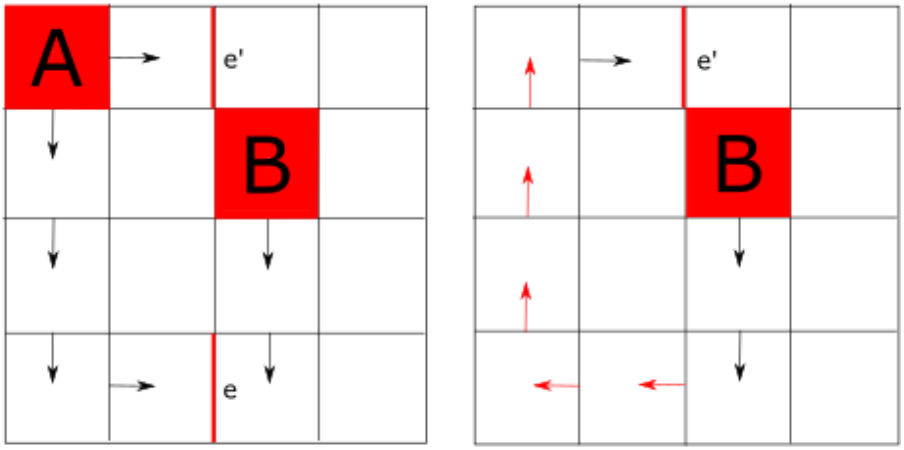

Figure 3.7: Adjustments in the discrete gradient vector field after cancellation of critical cells

3.3.6 Bifurcation diagrams

Bifurcation diagrams suggest how critical cells that indicate interesting features in the data are connected between time frames. These connections are obtained by putting time frames into a 3D cubical complex and extending the discrete gradient vector field from the slices to the whole complex following the gradient paths. Two critical cells in two neighboring slices are connected in the bifurcation diagram, if there is a gradient path in the 3D structure connecting them. Similar work has been done in [17].

Suppose we have a sequence of images each taken at different times

$$t_0 < t_1 < ... < t_n$$

We define a discrete Morse function F_{t_i} on each of the images and determine its discrete gradient vector field V_i. We think of the function F_{t_i} as a function that varies with time and we want to find a way to connect the critical cells from V_i to the ones in V_{i+1}. Following the definition in [17] if α and β are two critical k-cells in V_i and V_j (i $\neq$ j) respectively, we can say that α is connected to β if there is a k-cell γ and a V_i path α, ... , γ of k and $(k-1)$-cells and a V_j path γ, ..., β of k and $(k+1)$-cells.

Since we are only interested in 2-dimensional cells, we can conclude that a 2-dimensional critical cell α in V_i is connected with a 2-dimensional critical cell β in V_j if there is a 2-dimensional path in V_i which goes through α and ends in β. The algorithm we implemented does exactly that. Given a sequence of two images taken at time t and $(t-1)$ with discrete gradient fields V_i and V_{i-1} respectively, it goes through all valid 2-dimensional gradient paths in V_i and examines whether there is a 2-dimensional critical cell from V_{i-1} along that path. If there is such path, those two critical cells are connected and are added in the bifurcation diagram. The comparison between the two critical cells is done by their (x, y) coordinates (their offset in the image raster). Figure 3.8 shows how this algorithm connects critical cells between time frames. We can see that critical cells B and C at time $(t-1)$ are both connected with the critical cell A at time (t) because there

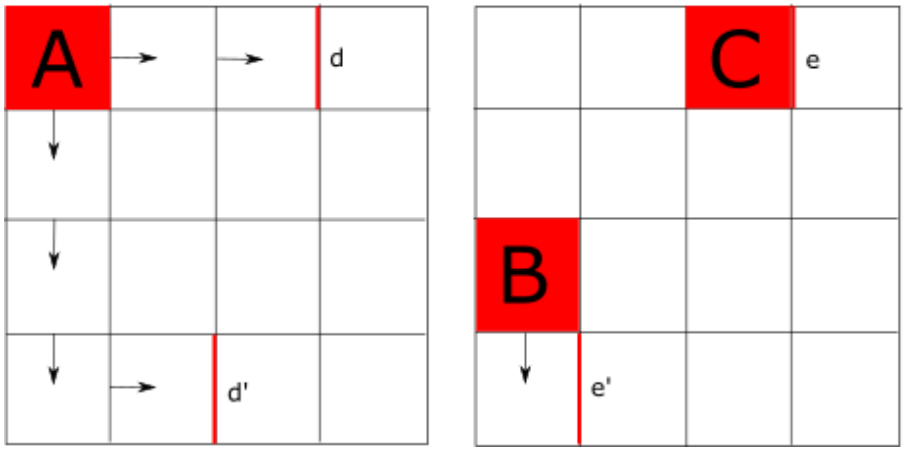

Figure 3.8: The image on the left shows the discrete vector field at time $(t-1)$ and the image on the right shows the discrete gradient field at time t

is a path from A that contains B and C.

When the data set represents a sequence of meteorological images, bifurcation diagrams can be seen as a tool for extracting relevant portions of the image where the value of the function (reflectivity) is the highest. By exploring a one hour long sequence (up to 6 meteorological images in our case), we can follow the way these portions interact between each other and make predictions based on certain behaviors that might appear in the bifurcation diagrams. In case of severe weather phenomena, we expect these portions to integrate into bigger structures or disintegrate in case of better weather conditions. Integration into bigger structures would mean that in case of a bad weather, 2-dimensional critical cell at time t is connected with a great number of 2-dimensional cells at time $(t-1)$.

Chapter 4

Results and discussion

In this chapter we present the obtained results.

4.1 Detecting important features in images

As explained earlier, in Section 3.3.2, the discrete gradient vector field helps with extraction of important features from a given data set. In our example, the data set is represented by a sequence of images, where each image is 401px wide and 301px high. After applying the above mentioned algorithms on each image separately, we got many critical cells and gradient paths. Most of these critical cells were local minima and saddles and were not of great use to us. By applying filtering with a certain threshold ϵ, we managed to decrease the number of critical cells and gradient paths significantly. We then took into account only the 2-dimensional critical cells (maxima) and the gradient paths that lead from 2-dimensional critical cells to the 1-dimensional critical cells.

It is very important to choose the right threshold ϵ, since choosing the wrong threshold can result in non-inclusion of essential features in the image. In this master thesis, we decided to set the threshold to 40 dBZ, because we suppose that 40 dBZ is the lowest reflectivity value that can be associated with bad weather. According to Table 3.1, the value of 40 dBZ means "Mod-

Image	Critical cells in the Morse Complex	Critical cells after filtration with 40 dBZ	Critical cells after filtration with 45 dBZ	Critical cells after filtration with 50 dBZ
1	45 820	86	67	25
2	45 974	70	41	9
3	41 713	63	17	0
4	47 222	119	72	11
5	45 887	81	31	1

Table 4.1: Number of 2-dimensional critical cells obtained after the construction of the Morse complex and then filtered by different types of thresholds (in dBZ)

erate rain", but since we are testing our method on a one hour long sequence, we must take into account that sometimes even lower reflectivity values can start a process that can later lead to severe weather conditions.

Image	Critical cells in the Morse Complex	Critical cells after filtration with 40 dBZ	Critical cells after cancellation
1	45 820	86	43
2	45 974	70	35
3	41 713	63	47
4	47 222	119	77
5	45 887	81	44

Table 4.2: Number of 2-dimensional critical cells obtained after the construction of the Morse complex, application of $\epsilon = 40$ dBZ threshold and then cancellation

Table 4.1 shows how the number of 2-dimensional critical cells changes

when running the Morse complex algorithm (Section 3.3.3) with different values for the ϵ threshold. The "Morse Complex" column contains the number of 2-dimensional critical cells obtained after applying the algorithm presented in Section 3.3.3 on a single image. The columns "40 dBZ", "45 dBZ" and "50 dBZ" show the number of 2-dimensional critical cells obtained after a filtering with threshold of 40 dBZ, 45 dBZ and 50 dBZ, respectively.

Table 4.2 shows the number of 2-dimensional critical cells after application of the cancellation theorem. We can see that the number of 2-dimensional critical cells is notably lower than the one we had before the cancellation. It is also worth noting that most of the cancelled critical cells were those for which there was a gradient path between the critical cell and its face. Such path represents a gradient path between a 2-dimensional critical cell and one of its 1-dimensional faces that is also critical. In cases like this, the changes that appear in the function are hardly noticeable, so it is normal for those cells to be cancelled.

The black rectangles in Figure 4.1 represent the 2-dimensional critical cells, that were obtained after applying the algorithm for construction of the Morse complex (Section 3.3.3) and filtering with a threshold $\epsilon = 40$ dBZ. Figure 4.2 shows the 2-dimensional critical cells after applying the cancellation theorem. The above mentioned images refer to Image 5 in Table 4.2.

The cancellation algorithm was also applied in case of existance of gradient paths that connect critical cells, such that the value of the 2-dimensional starting cell is insignificantly higher than the 1-dimensional ending cell. As in the previous example, these cases mean small changes in the function, so these types of critical cells must not be taken into account. This type of filtering requires definition of a new threshold. We set this threshold to 1, because in our set of images, we presume that 1 is the lowest value that can depict changes in the function.

As explained earlier, the cancellation algorithm requires comparing the values of a 2-dimensional critical cell (rectangle) and a 1-dimensional critical

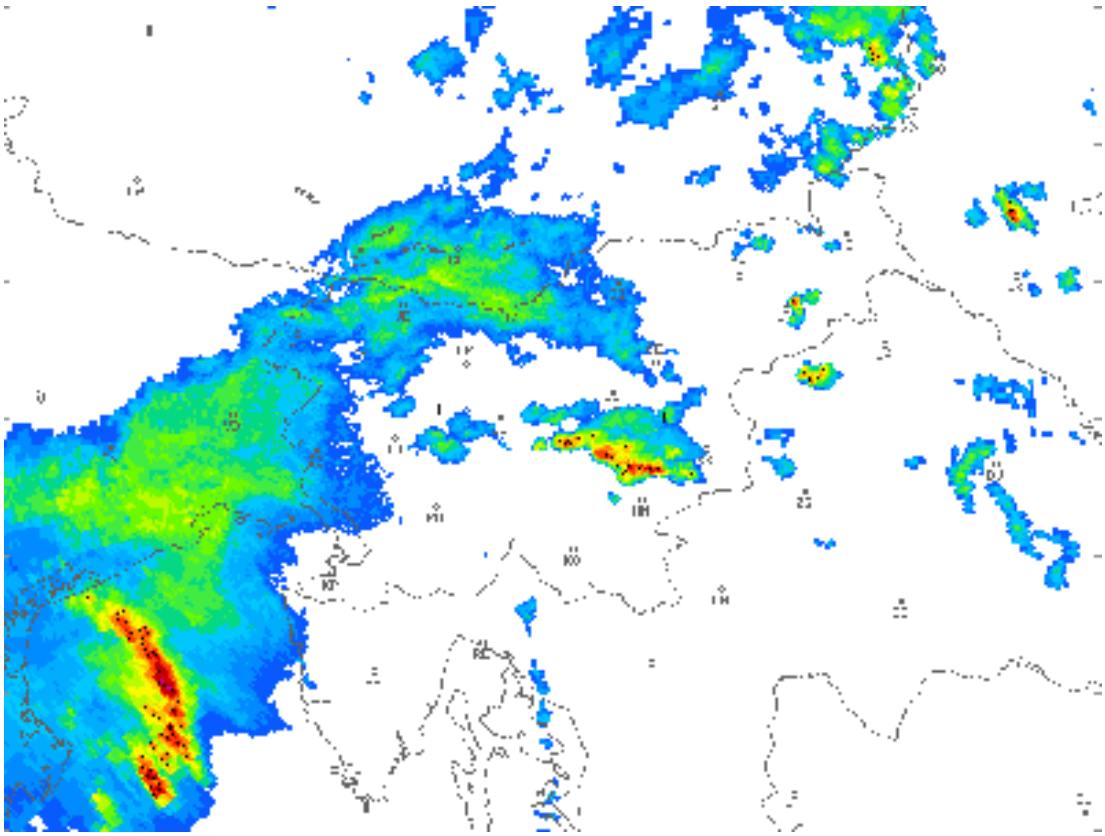

Figure 4.1: 2-dimensional critical cells obtained after construction of the Morse complex and filtration with a 40 dBZ threshold

cell (edge). For the purpose of the algorithm, as values for the 2-dimensional critical cell and the 1-dimensional critical cell, we used the maximum value of the four points consisting the 2-dimensional critical cell and the maximum value of the two points forming the 1-dimensional critical cell, respectively.

4.2 Bifurcation diagrams

The major goal of this master thesis is the construction of bifurcation diagrams. As stated earlier, bifurcation diagrams can help in building connections between important features in the images through different time frames.

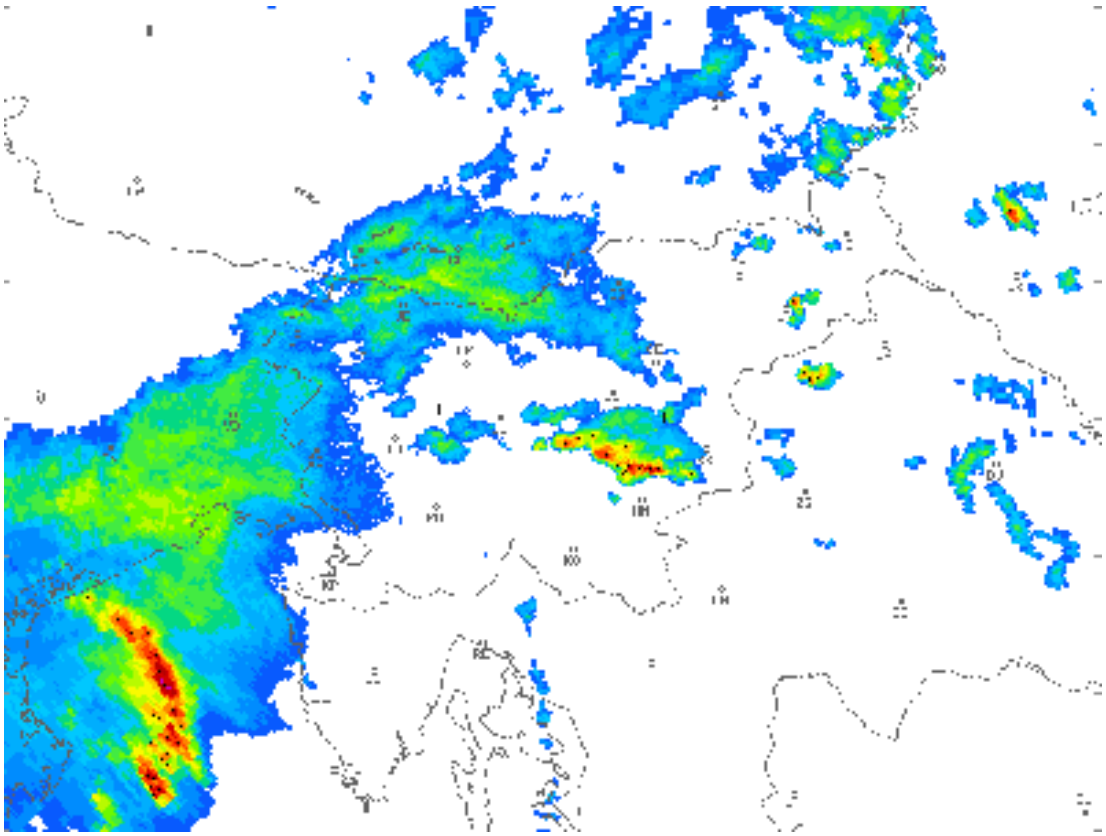

Figure 4.2: 2-dimensional critical cells obtained after cancelling

Since we also work with a sequence of images taken at different periods of time, we expect that bifurcation diagrams can be applicable to our problem as well.

Severe weather conditions (heavy rain, hail, etc.) can be observed on the images as bigger structures where the value of the function (the reflectivity) reaches up to 50 - 52 dBZ (sometimes even up to 60 dBZ in the case of a hail). Therefore, the information that we would like to obtain from the bifurcation diagrams in our case will show how certain structures organize into bigger ones.

We examine a one hour long sequence of meteorological images. This means that we have 6 meteorological images taken at a 10 minute interval.

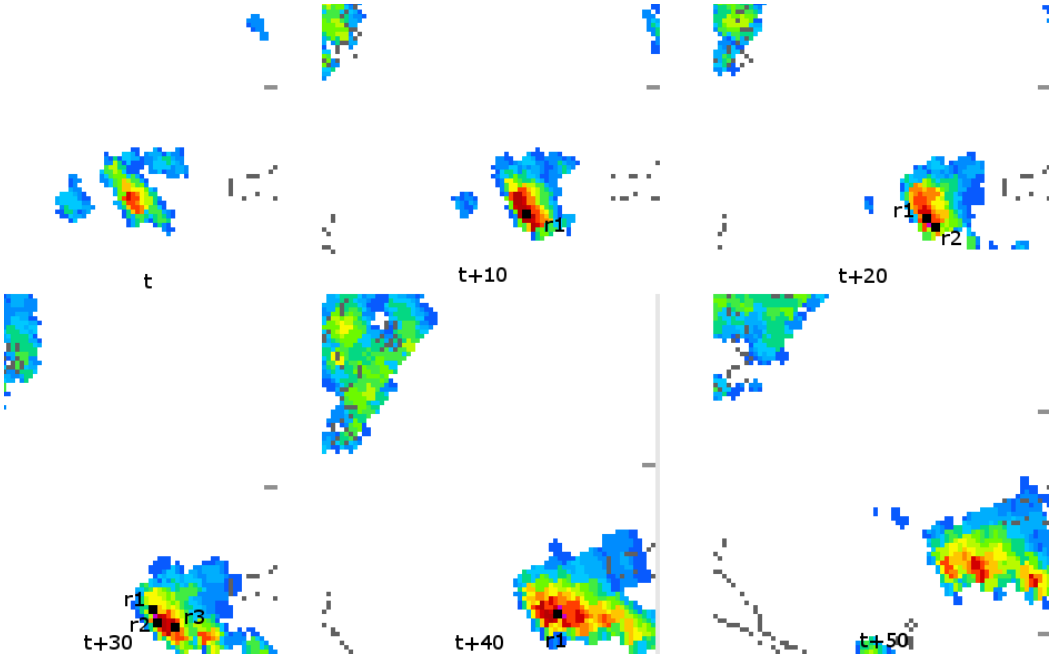

Figure 4.3: 2-dimensional critical cells obtained at time t, $t + 10$, $t + 20$, $t + 30$, $t + 40$, $t + 50$ (from left to right)

Each bifurcation diagram connects an image taken at time t with the previous image taken 10 minutes earlier. For example, if we have 6 images, the algorithm will produce 5-stage bifurcation diagram, each stage connecting a given image with its predecessor, until the algorithm gets to the last (most recent) image. For practical reasons, we set the length of the sequence to one hour and we suppose that a one hour time frame is enough to track down certain critical features and follow their interactions that later might lead to bad weather conditions in a given area. However, longer time frames would most likely be necessary for more practical weather prediction.

In order to be able to explain how bifurcation diagrams work and how we can search for interesting interactions in them, here we present the results we obtained by building bifurcation diagrams for two different sequence of images. The first sequence of images is given in Figure 4.3 and the second sequence of images is given in Figure 4.5. Since the bifurcations diagrams

are quite long, here we only present interesting parts of both bifurcation diagrams which demonstrate how certain structures compose into bigger ones or decompose into smaller ones through time.

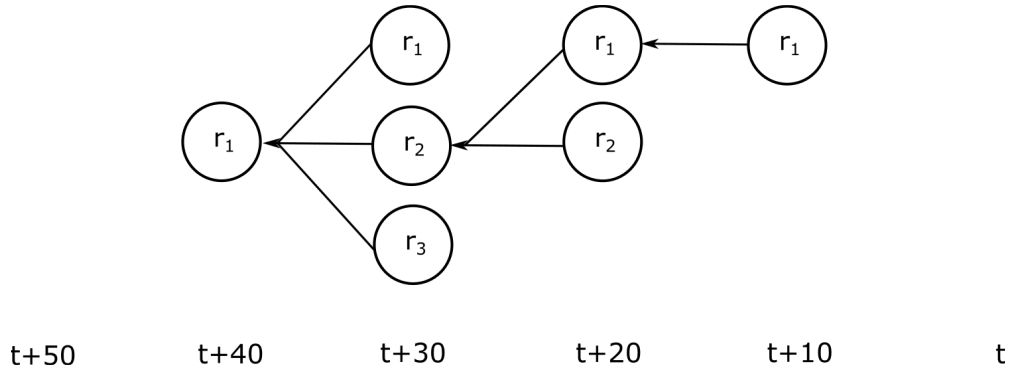

Figure 4.4: The bifurcation diagram for the sequence of images in Figure 4.3

If we take a look at Figure 4.3 we can see a sequence of 6 meteorological images. Each image represents a part of a bigger image. As stated previously, bifurcation diagrams can be quite long, so we decided to concentrate only on those parts of the image for which the algorithm returned interesting features. 2-dimensional critical cells at a certain time segment are represented with black rectangles. For viewing purposes, the width and the height of the black rectangles are bigger than the real ones and differ by 1. At each time segment, we only show the 2-dimensional critical cells for which there is a connection in the bifurcation diagram at the next time segment. For example, if we take a look at Figure 4.4 we can see that at time t there are no 2-dimensional critical cells shown in the image. That happens because none of the 2-dimensional critical cells that were detected at time t are connected with any 2-dimensional critical cells detected at time $t + 10$. It might appear strange that connections between 2-dimensional critical cells from different time segments do not exist, but such situations can happen because of the dynamics in the weather system. Since in Figure 4.3 are given only parts of the actual weather images, it is hard to see that the structure represented at time t has shifted to the right at time $t + 10$. Because of the shifting,

no 2-dimensional critical cells detected at time t are connected with the 2-dimensional critical cells detected at time $t + 10$. Another reason for such behaviour can be the fact that there are no 2-dimensional critical cells with values above the threshold, so connections between different stages in the bifurcation diagram won't be possible.

The bifurcation diagram in Figure 4.4 shows how the critical cells drawn in Figure 4.3 are connected to each other. We start looking at the bifurcation diagram at time t. As stated, there are no 2-dimensional cells at this time, so the first 2-dimensional cell to appear is at time $t + 10$. At time $t + 20$ there are two 2-dimensional critical cells. The first one is connected to the 2-dimensional cell detected at time $t + 10$ whereas the other 2-dimensional cell was born at time $t + 20$. Then both 2-dimensional critical cells detected at $t + 20$ connect to a new 2-dimensional critical cell detected at $t + 30$. At time $t + 30$ together with two other 2-dimensional critical cells connect to the 2-dimensional critical cell at $t + 40$ and result into creating the bigger structure given in the before last image in Figure 4.3. Finally, at $t + 50$ we can see that again, there are no 2-dimensional critical cells. This happens because none of the 2-dimensional critical cells at $t + 40$ connect to any other 2-dimensional critical cells detected at time $t + 50$. This features can be also seen in the last image from Figure 4.3, namely the big structure created at $t + 40$ has decomposed.

Such and similar types of bifurcation diagrams, where more 2-dimensional cells are connected to a single 2-dimensional cell (n $\rightarrow$ 1 connection), can provide us with a hint about bad weather conditions. As explained earlier, severe weather phenomena can be detected in the images as bigger structures where the value of the function (the reflectivity) reaches up to 50 - 52 (sometimes even more) dBZ. The creation of n $\rightarrow$ 1 connections through the sequence of images show how those big structures are created.

Another sequence of images is given in Figure 4.5. Similar to the sequence given in Figure 4.3, the second sequence also contains 6 different meteorological images each taken at different time segments. The bifurcation diagram

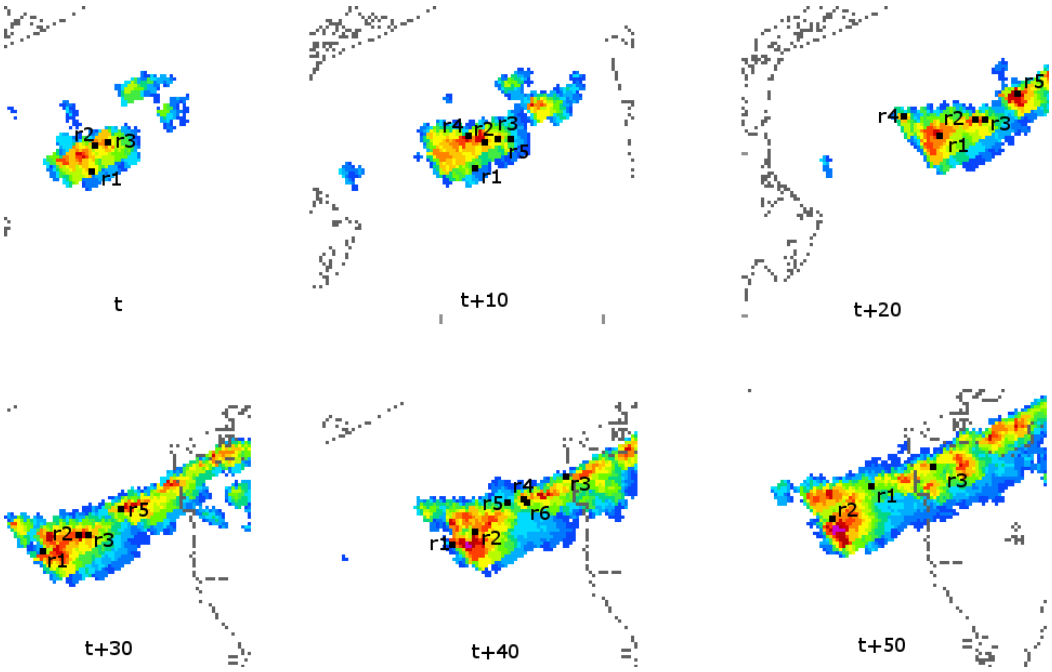

Figure 4.5: 2-dimensional critical cells represented with black rectangles, that were obtained at time t, $t + 10$, $t + 20$, $t + 30$, $t + 40$, $t + 50$ (from left to right)

for the second sequence is given in Figure 4.6. The 2-dimensional cells at each time segment are presented as black rectangles denoted with r_1, r_2, r_3, r_4 and so on. Bear in mind that, r_i where $i = 1, 2, 3, \ldots$ n at time $t + t_0$ where $t_0 = 10,.. 50$, are not the same. They can be, but not necessarily.

If we take a look at the bifurcation diagram for Figure 4.5, we can see that most of the connections here are $1 \to 1$ connections that do not last long. They are usually one or two time frames long. We can also conclude the same if we take a look at the sequence of images in Figure 4.5. We can see that there is no composition into bigger structures, new 2-dimensional critical cells are born each time frame, they can make $1 \to 1$ connection with another 2-dimensional critical cell in the next time frame, but such connection lasts short period of time. There is an attempt for composition into a bigger structure around r_1 at time $t + 20$ as a result of the conjugation

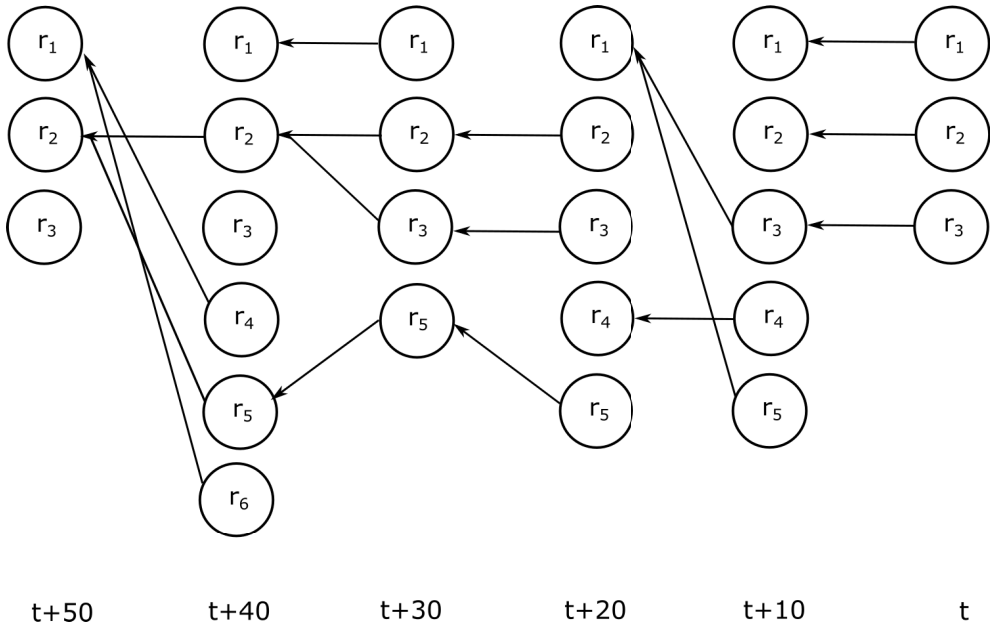

Figure 4.6: The bifurcation diagram for the sequence of images in Figure 4.5

of r_3 and r_5, but at time $t + 30$ this structure is already decomposed and new 2-dimensional critical cells are born. Except for the bigger structure forming around r_2 at time $t + 40$, which started building up at time $t + 30$ and continued to $t + 40$, we can conclude that all other connections denote either decomposition either no sign of changes in the weather.

Such and similar types of bifurcation diagrams, where one 2-dimensional cell is connected to a single 2-dimensional cell ($1 \to 1$ connection), can provide us with a hint about more favorable weather conditions since smaller structures do not concentrate into bigger ones in this case.

4.3 Classification of bifurcation diagrams

As mentioned earlier, we constructed bifurcation diagrams for a one hour sequence and tried to find correlations between the bifurcation diagrams and the weather conditions presented in the images. In addition, we searched for interesting features that might lead to bad weather conditions and tried to

determine when and where such features appear.

The two examples of bifurcation diagrams given in Figure 4.4 and 4.6, lead us to the conclusion that we can use the number of entry connections as a parameter for classification of the bifurcation diagrams. In this master thesis, we refer to the number of entry connections as "in-degree". That being said, if we take a look at Figure 4.4, we can determine the "in-degree" of the given 2-dimensional critical cells. The "in-degree" of r_1 at time $t + 40$ is equal to 3, the "in-degree" for r_2 at time $t + 40$ is equal to 2, etc.

ID	Maximum dBZ	$1 \rightarrow 1$ connections	$2 \rightarrow 1$ connections	$n \rightarrow 1$ connections, $n > 2$	Sum($2 \rightarrow 1$, $n \rightarrow 1$)
1	45	1	1	0	1
2	57	53	20	5	25
3	54	144	28	13	41
4	57	38	10	4	14
5	57	69	16	3	19
6	57	59	8	2	10
7	57	63	19	5	24
8	57	29	8	2	10
9	57	57	7	2	9
10	51	74	13	3	16
11	57	79	32	5	37
12	48	51	9	1	10
13	54	59	3	2	5
14	51	55	10	1	11
15	54	158	36	11	47

Table 4.3: Computation of maximum reflectivity value and in-degree in a 5-stage bifurcation diagram

Table 4.3 shows the number of $1 \rightarrow 1$, $2 \rightarrow 1$, $n \rightarrow 1$ ($n > 2$) connections

for 15 different 5-stage bifurcation diagrams obtained for a one hour long sequence. The column "Maximum dBZ" represents the maximum reflectivity value reached in a single 5-stage bifurcation diagram, the column "$1 \to 1$ connections" represents the number of $1 \to 1$ connections ("in-degree" equals 1), the column "$2 \to 1$ connections" represents the number of $2 \to 1$ connections ("in-degree" equals 2), the column"$n \to 1$ connections, $n > 2$" represents the number of $n \to 1$ connections ("in-degree" equals more than 2) and finally, the column "Sum($2 \to 1$, $n \to 1$)" represents the sum of $2 \to 1$ and $n \to 1$ connections.

We tried to use the results in Table 4.3 to build a simple model with which we can classify the bifurcation diagrams, so when a new bifurcation diagram comes in, we can find its place in the right group: possibility of severe weather or no such possibility. For classifying the bifurcation diagrams we used hierarchical clustering with the help of Orange [14]. We used complete linkage with Manhattan distances, where the classification parameter is the sum of $2 \to 1$ and $n \to 1$ connections for a single 5-stage bifurcation diagram. The reason we include only the $2 \to 1$ and $n \to 1$ connections in the classification is that, $1 \to 1$ connections are not of great use to us and they do not represent changes in the bifurcation diagrams, nor important features in the sequence of images.

In Figure 4.7 is shown the dendogram we obtained for the results in Table 4.3. We can see that it identifies two clusters. The red cluster identifies the bifurcation diagrams in which the sum of entry connections is higher indicating a possibility of severe weather conditions. The labels in the dendogram represent the column "ID" in the Table 4.3.

The results in Figure 4.7 are just a simple example of how the number of entry connections in a 5-stage bifurcation diagram can be used for further exploration of the bifurcation diagrams. The number of $n \to 1$ ($n \geq 2$) connections is significantly higher in the case of bad weather conditions, than in the case of better weather conditions. Again, this result is due to the fact that critical cells join into bigger structures when there is a possibility

of severe weather. However, there are drawbacks of the classification model presented in this section. We discuss these drawbacks and present ideas for improvement in Section 5.2.

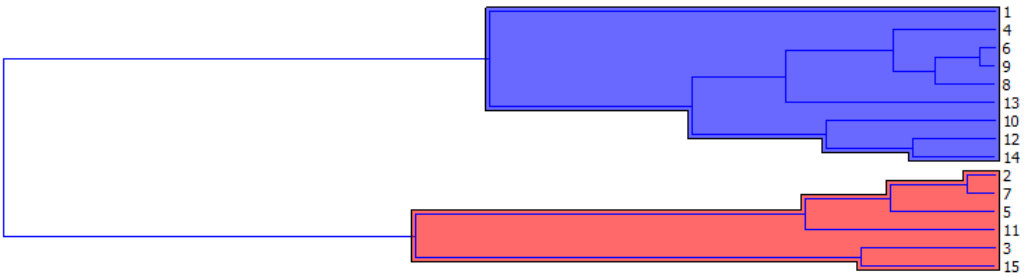

Figure 4.7: The obtained dendogram after the classification of the results obtained in Table 4.3

4.4 Discussion

We are well aware that weather is a huge and complicated process and that is dependent on many variables. In this master thesis we concentrated solely on the precipitation intensity and its impact on weather forecasting. We based our method on TDA and we obtained promising results that have the potential to be later used as a reference in future researches. We also showed that TDA can be used in solving problems similar to the one described here and that it can produce promising results.

As described earlier, weather is a very dynamic process and the results of the methods applied above can be greatly influenced by other parameters such as: air temperature, atmospheric (barometric) pressure, humidity, wind and others. Non-inclusion of such parameters and the dynamics of the system can especially cause problems in the construction of the bifurcation diagrams. Such problems can occur, since we construct the bifurcation diagrams to be

dependent on the geographic coordinates of the region (or the x and y offset in the image). This type of implementation combined with the constant changes, the shifting and the dynamics of the weather system can sometimes result in loss of important features and connections between the images.

In Section 5.2 we discuss some of the drawbacks of the used algorithms and the way they can be improved in the future.

Chapter 5

Conclusion and future work

5.1 Conclusion

In this master thesis we concentrated on the dynamics of changes in the precipitation intensity through a time sequence of radar images, and inspected how this can be used as a parameter in automatic severe weather prediction. We used methods of topological data analysis, in particular discrete Morse theory. The data set on which we based our research was provided by ARSO and was represented by sequences of meteorological images taken at different time intervals.

Our main idea was to extract important features in each image and to try and connect those features between different time frames, so that we can make weather predictions. We examined each image separately and represented it as a cubical cellular complex. We then defined a Morse function on each cubical complex and obtained the discrete gradient vector field of the function. The construction of the gradient vector field helped us in the search of critical cells. In the next step we built the Morse complex and obtained paths that connect the obtained critical cells. At the end, we tried to expand those paths through the whole sequence of images and search for connections between critical cells in different time frames. We managed to succeed this, by building bifurcation diagrams.

We also gave a simple example of how the connections in the bifurcation diagrams can be used to track changes in the weather system and presented a model for classification of the bifurcation diagrams.

5.2 Future work

There is always a possibility and room for improvements, so in this section we give a few ideas about what can be done better with regard to the methods we used in this master thesis and how we can use the potential of the obtained results as a basis for further researches in the field of weather forecasting.

In this master thesis we concentrated on the dynamics of changes in the precipitation intensity through a time sequence of radar images and inspected how this can be used as a parameter in automatic severe weather prediction. That being said, we had to concentrate only on the precipitation intensity and define our Morse function to be dependent only on the reflectivity (precipitation). In the future, it will be useful to take into account other parameters that are known to influence weather predictions. These additional parameters can be used when defining the Morse function, so that it is no longer dependent only on the precipitation, but also on air pressure and wind. This is essential, because the algorithm can fail to connect important features in the images, caused by the fact that some of the gradient paths can be lost as a result of wind, or other atmospheric change that might appear between different time frames.

There is an interesting paper [18] on gathering useful meteorological data from aircraft using Mode-S radars. It is mostly concentrated on obtaining meteorological data (air pressure, temperature and winds), but also gives an insight into the most used parameters in weather forecasting and how they can influence short term forecasts. Using the additional data obtained in this way combined with the proposed data analysis methods would provide additional insight into the weather and increase prediction accuracy.

In this master thesis, we only considered an hour long sequences of images,

due above all to the limited resources (computer power) and time availability. Our preliminary results are promising, even though, taking into account longer sequences of images might lead to different and probably even better results.

Another thing that we would like to address is the classification of the bifurcation diagrams. We managed to create a simple model based on the sum of the $n \to 1, n \geq 2$ connections of all nodes. However, this parameter might not be enough in some situation and we must include other parameters in the classification. One additional parameter that can be included in the classification is the precipitation intensity. We can use the precipitation intensity to filter out connections that exist between critical cells with intensity values smaller than some threshold.

There is also another possibility of classification. If we look at the set of bifurcation diagrams obtained for a single sequence of images, we can represent the set as directed weighted graph. This can allows us to use already known algorithms for weighted directed graphs and acquire even better metrics for comparing graphs. The most suitable methods and algorithms would be those that offer evaluation of the similarity between two general directed weighted graphs of different dimensions, because each set of bifurcation diagrams can have a different dimension. Similar work has been done in [19].

Bibliography

[1] R. Forman. A user's guide to discrete morse theory. In *Proc. of the 2001 Internat. Conf. on Formal Power Series and Algebraic Combinatorics, A special volume of Advances in Applied Mathematics*, page 48, 2001.

[2] R. Forman. Morse theory for cell comlexes. *Adv. in Math 134*, pages 90–145, 1998.

[3] M. J. Atallah and M. Blanton. *Algorithms and Theory of Computation Handbook, Second Edition, Volume 2*. CRC Press, Taylor and Francis Group, 2009.

[4] R. Ghrist. *Elementary applied topology*. 2014.

[5] G. Carlsson. Topology and data. *Bull. Amer. Math. Soc. 46*, pages 255–308, 2009.

[6] G. Jerše and N. Mramor Kosta. Tracking features in image sequences using discrete morse functions. *Proceedings 3rd International Workshop on Computational Topology in Image Context (CTIC 2010)*, 1:27–32, 2010.

[7] S. K. Das, B. Chanda, and D. P. Mukherjee. Prediction of cloud for weather now-casting application using topology adaptive active membrane. *Pattern Recognition and Machine Intelligence Lecture Notes in Computer Science*, 5909:303–308, 2009.

[8] J. Boissonnat and M. Teillaud. *Effective Computational Geometry for Curves and Surfaces*. Springer, 2010.

[9] V. A. Kovalevskiy. Finite topology as applied to image analysis. *Computer vision, graphics, and image processing*, 46:141–161, 1989.

[10] Intelligent perception. `http://inperc.com/wiki`. [Online; accessed 19-July-2015].

[11] M. Morse. The calculus of variations in the large. *American Mathematical Society Colloquium Publication 18; New York*, 1934.

[12] Wikipedia - dbz (meteorology). `https://en.wikipedia.org/wiki/DBZ_(meteorology)`. [Online; accessed 25-May-2015].

[13] Z - r relationships. `http://apollo.lsc.vsc.edu/classes/remote/lecture_notes/radar/conventional/ZR_relations.html`. [Online; accessed 19-July-2015].

[14] Orange. `http://orange.biolab.si/`. [Online; accessed 13-July-2015].

[15] V. Robins, P.J. Wood, and A. P. Sheppard. Theory and algorithms for constructing discrete morse complexes from grayscale digital images. *Pattern Analysis and Machine Intelligence, The IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pages 1646–1658, 2011.

[16] O. Delgado-Friedrichs, V. Robins, and A. P. Sheppard. Skeletonization and partitioning of digital images using discrete morse theory. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

[17] H. King, K. Knudson, and N. Mramor Kosta. Birth and death in discrete morse theory. *available at arXiv:0808.0051v1*, 2008.

[18] M. Hrastovac and F. Solina. Obtaining meteorological data from aircraft with mode-s radars. *IEEE Aerospace and Electronics Systems Magazine*, 28(12):12–24, 2012.

[19] Y. Xu, S. Salapaka, and C. Beck. A distance metric between directed weighted graphs. *52nd IEEE Conference on Decision and Control*, 2013.

Chapter 6

Appendix

6.1 Examples of bifurcation diagrams

6.1.1 Example 1.

In this example is given another 5-stage bifurcation diagram for six different radar images. Since the bifurcation diagram is too long to be presented in a single image, we decided to split it and presented it in two images (Figure 6.7 and Figure 6.8). Figure 6.1, Figure 6.2, Figure 6.3, Figure 6.4, Figure 6.5 and Figure 6.6 are the weather images for which the bifurcation diagram is constructed. When applying the "Cancellation Theorem" in this example, a threshold of 0.7 was used.

6.1.2 Example 2.

In this example is given another 5-stage bifurcation diagram for six different radar images. Since the bifurcation diagram is too long to be presented in a single image, we decided to split it and presented it in two images (Figure 6.15 and Figure 6.16). Figure 6.9, Figure 6.10, Figure 6.11, Figure 6.12, Figure 6.13 and Figure 6.14 are the weather images for which the bifurcation diagram is constructed. When applying the "Cancellation Theorem" in this example, a threshold of 1 was used.

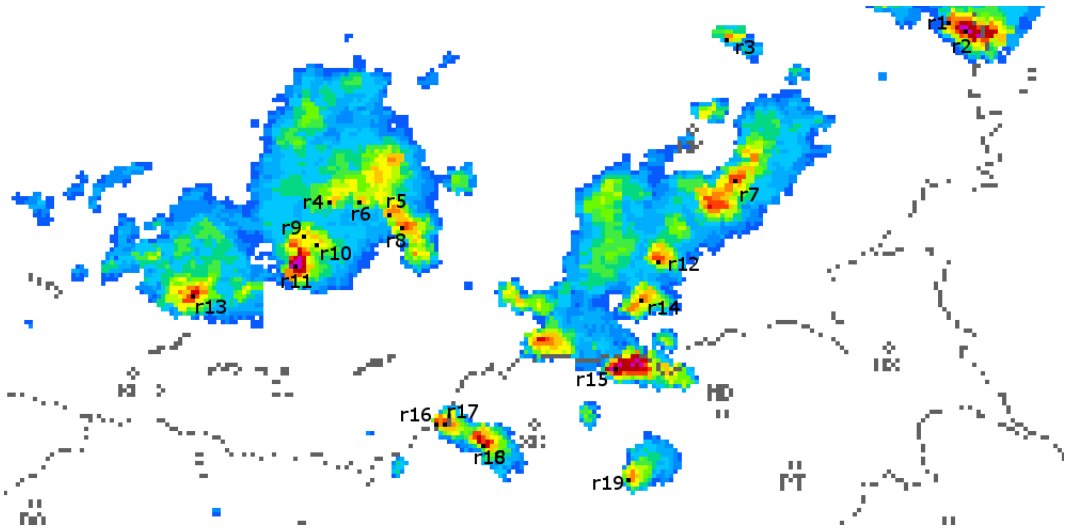

Figure 6.1: Example 1. Weather radar image taken at time t

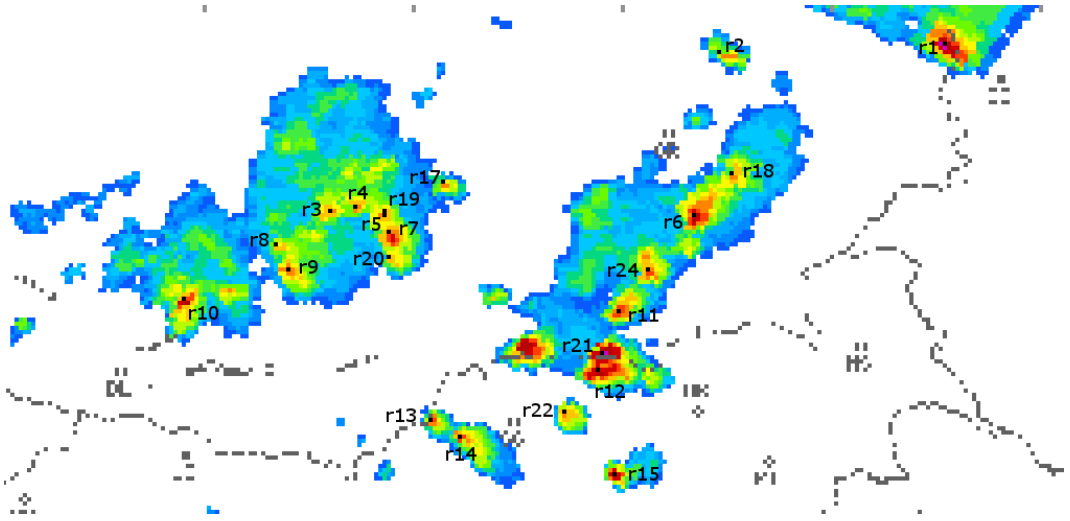

Figure 6.2: Example 1. Weather radar image taken at time $t + 10$

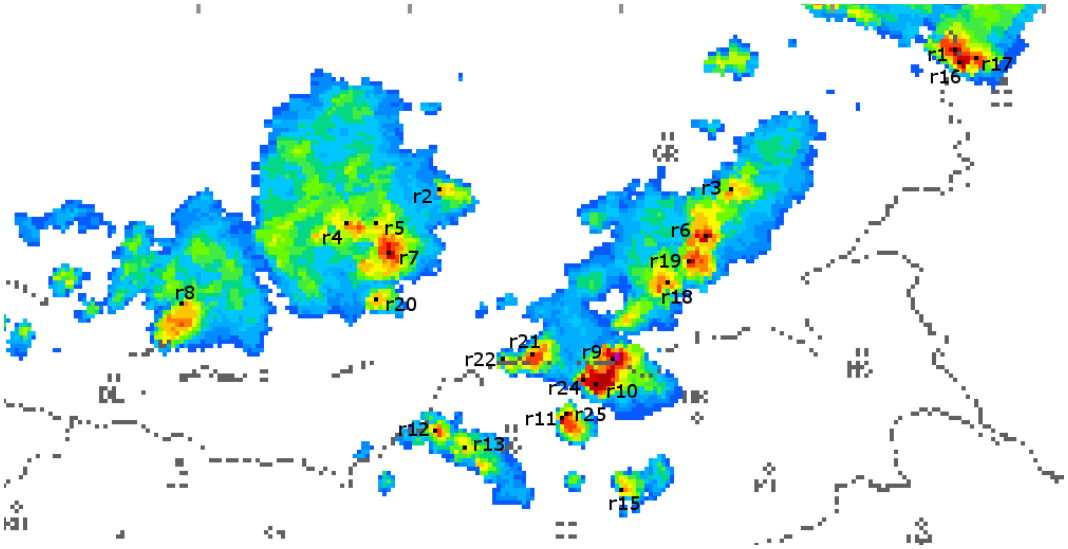

Figure 6.3: Example 1. Weather radar image taken at time $t + 20$

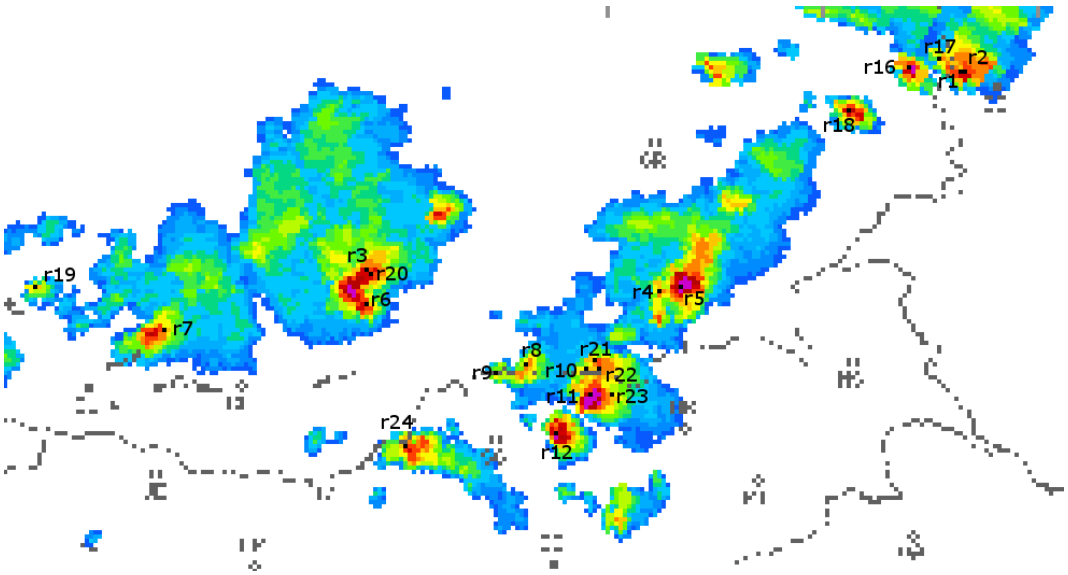

Figure 6.4: Example 1. Weather radar image taken at time $t + 30$

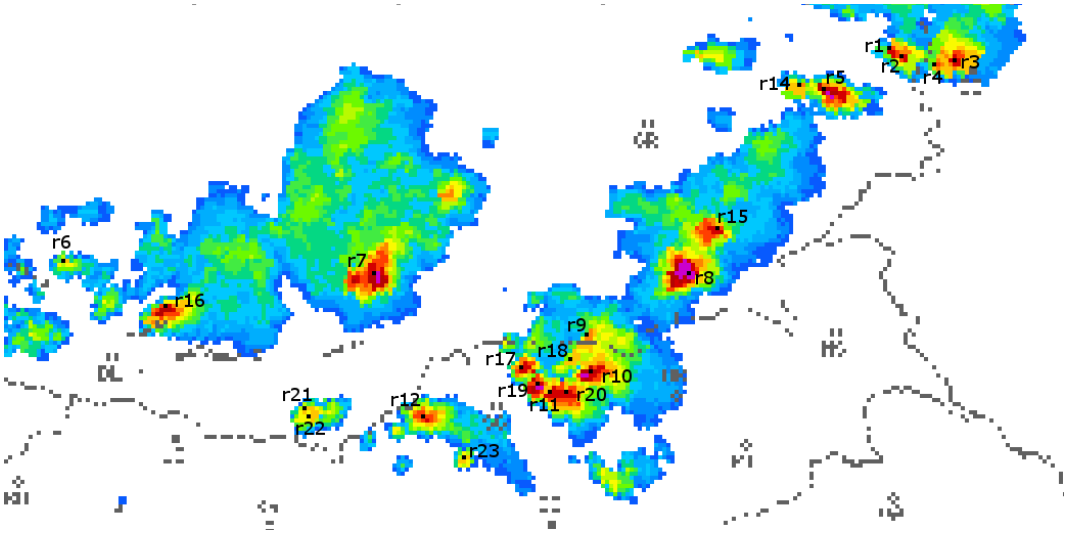

Figure 6.5: Example 1. Weather radar image taken at time $t + 40$

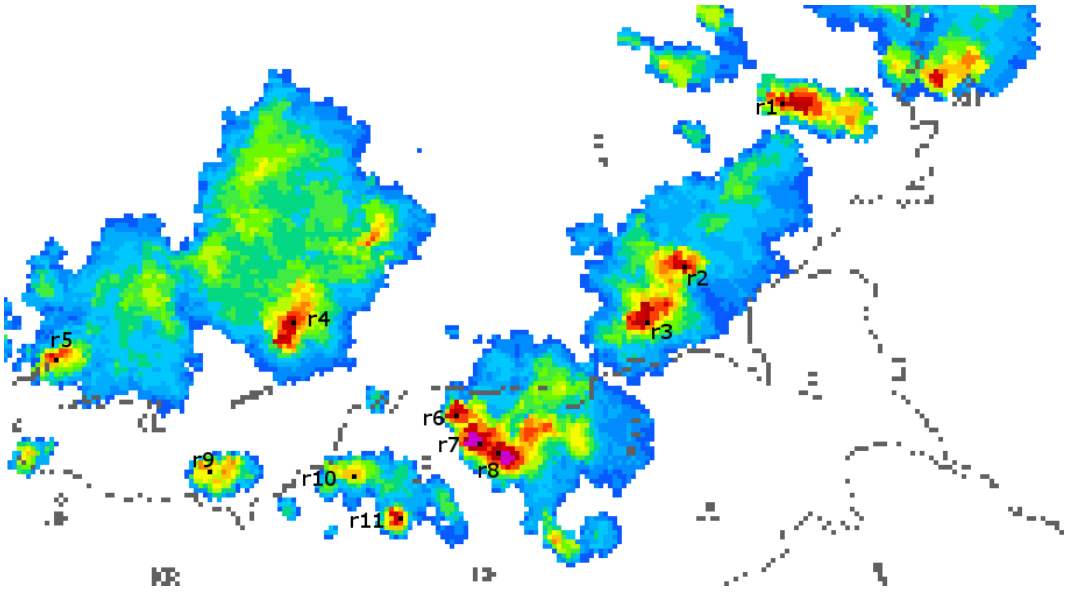

Figure 6.6: Example 1. Weather radar image taken at time $t + 50$

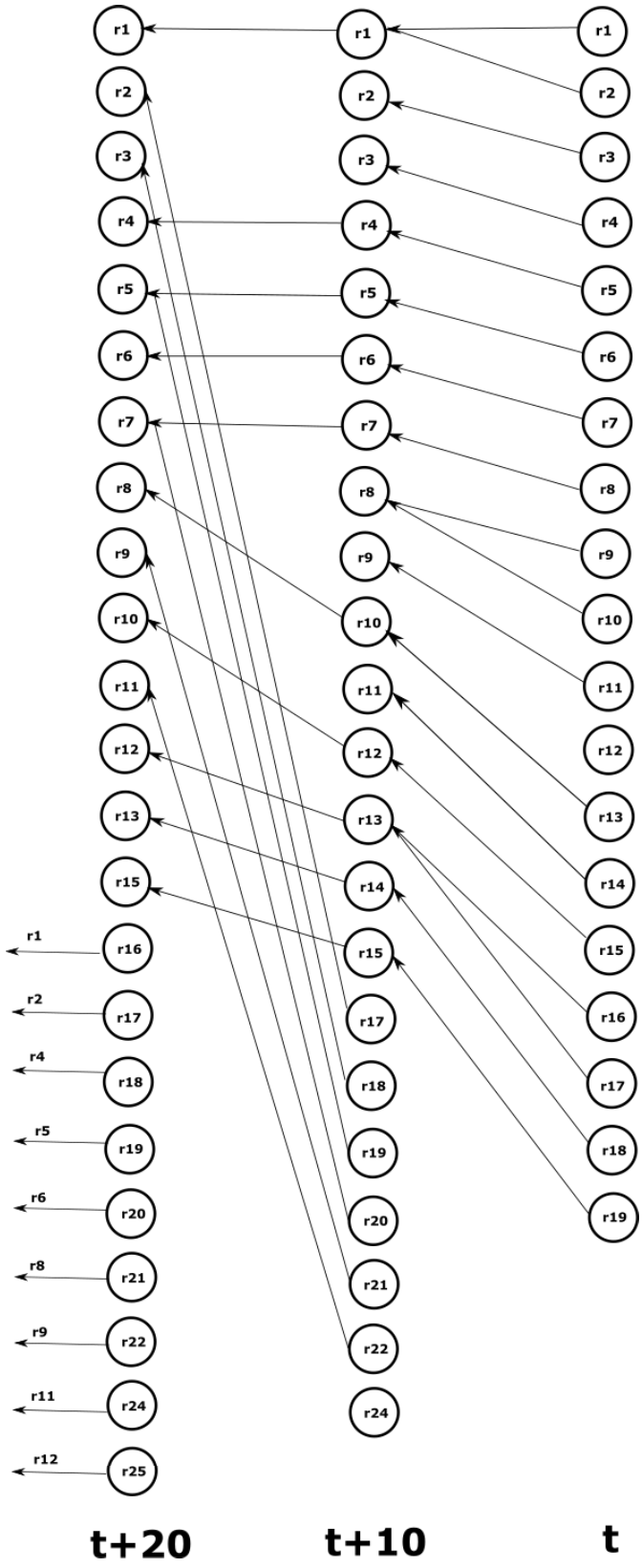

Figure 6.7: Example 1. The bifurcation diagram for the radar images taken at time t, $t + 10$, $t + 20$

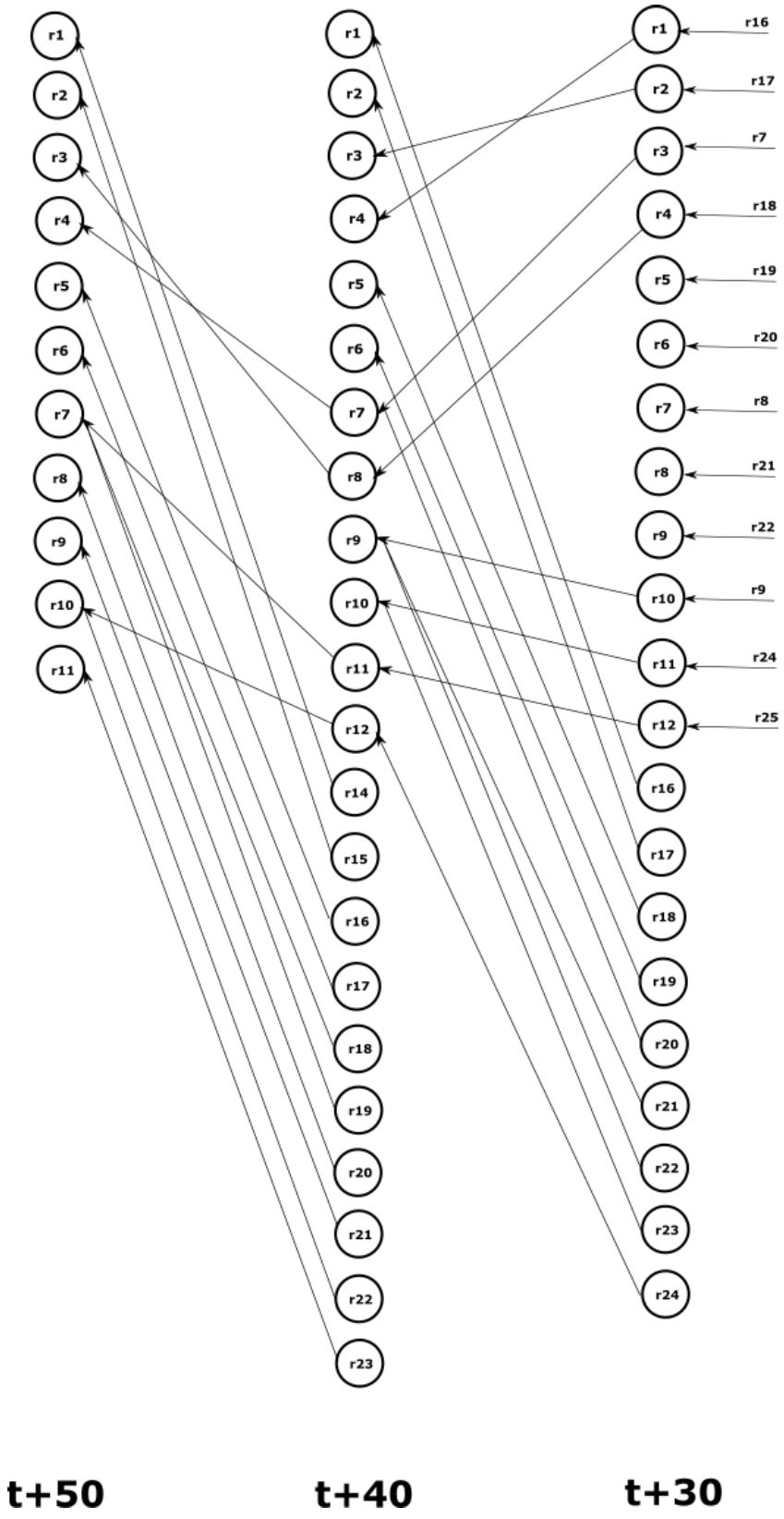

Figure 6.8: Example 1. The bifurcation diagram for the radar images taken at time $t + 30$, $t + 40$, $t + 50$

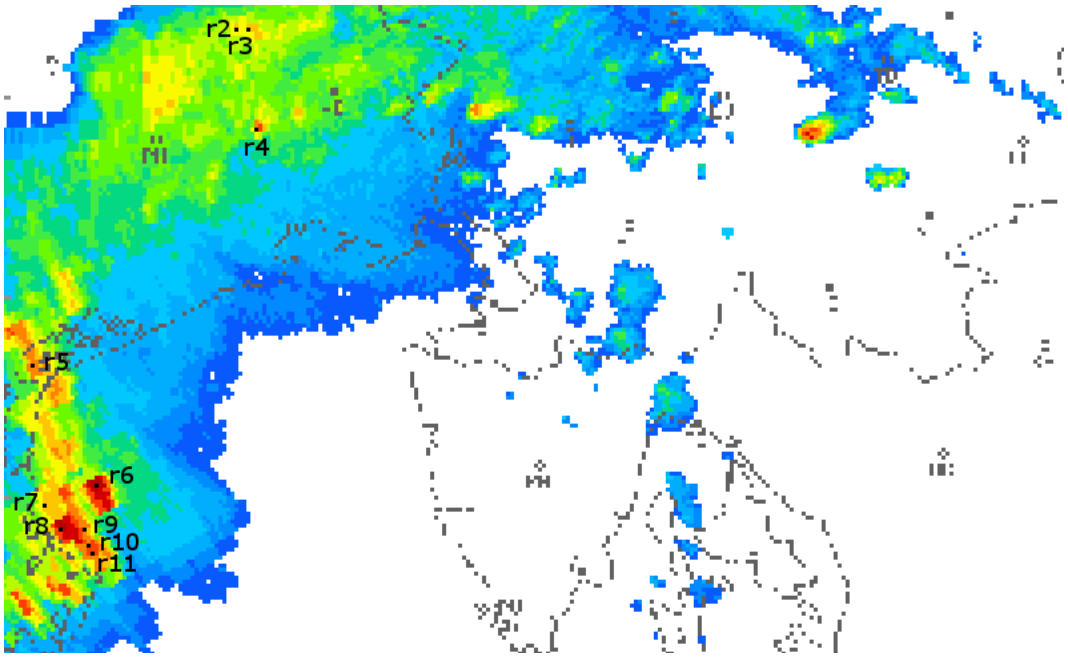

Figure 6.9: Example 2. Weather radar image taken at time t

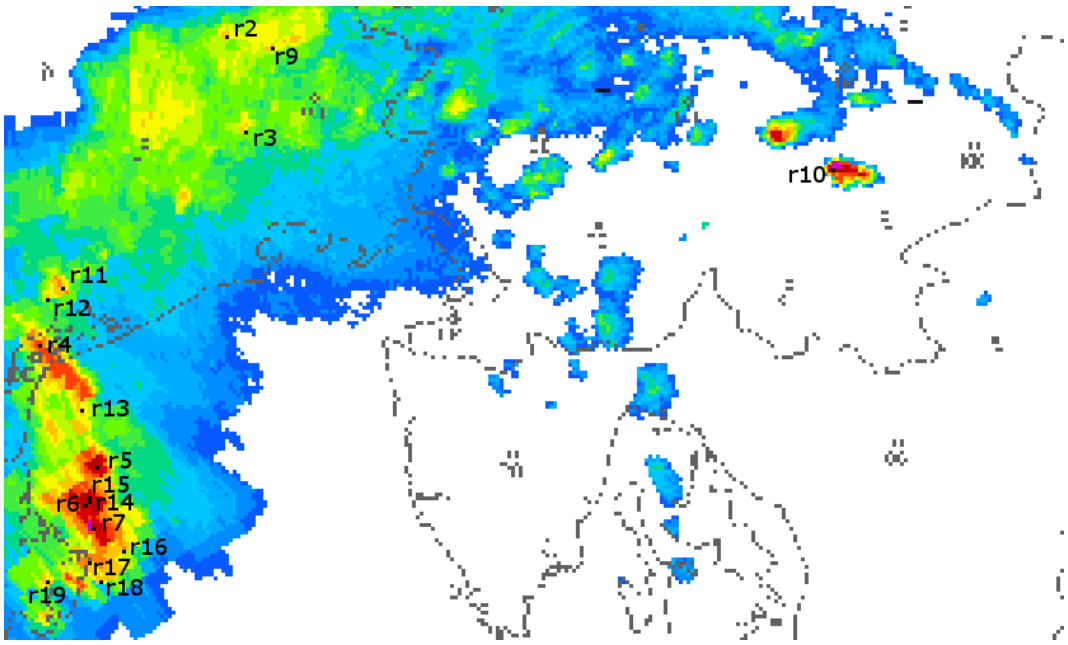

Figure 6.10: Example 2. Weather radar image taken at time $t + 10$

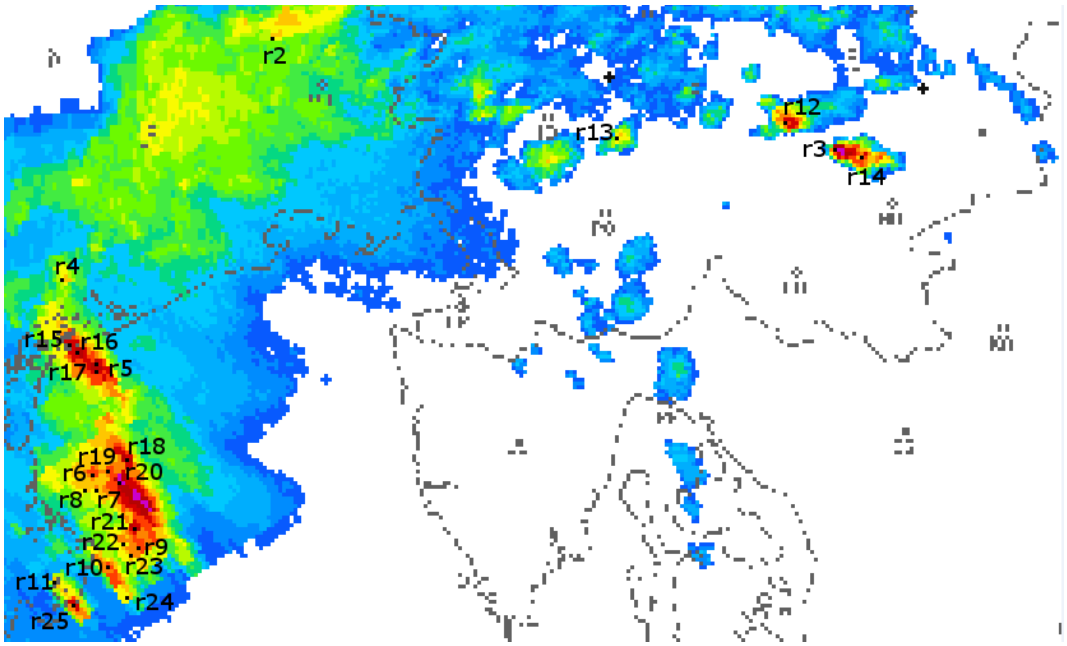

Figure 6.11: Example 2. Weather radar image taken at time $t + 20$

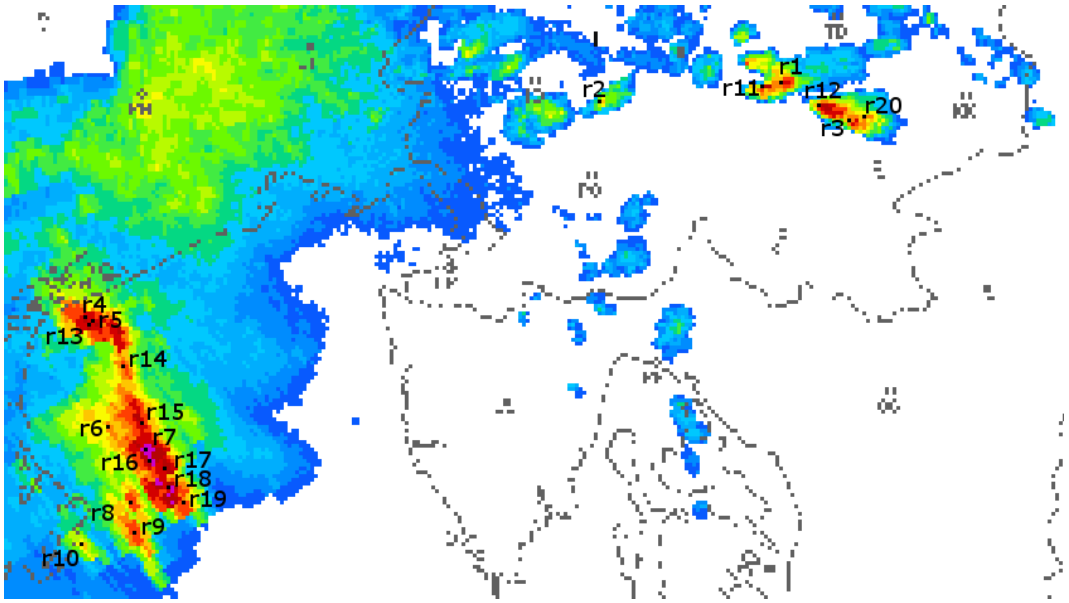

Figure 6.12: Example 2. Weather radar image taken at time $t + 30$

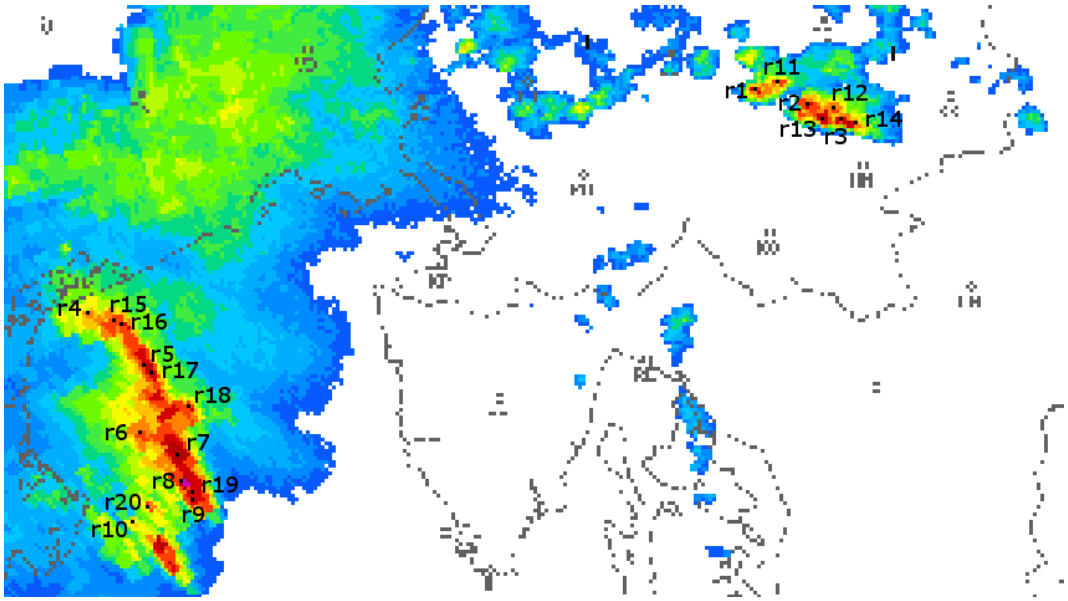

Figure 6.13: Example 2. Weather radar image taken at time $t + 40$

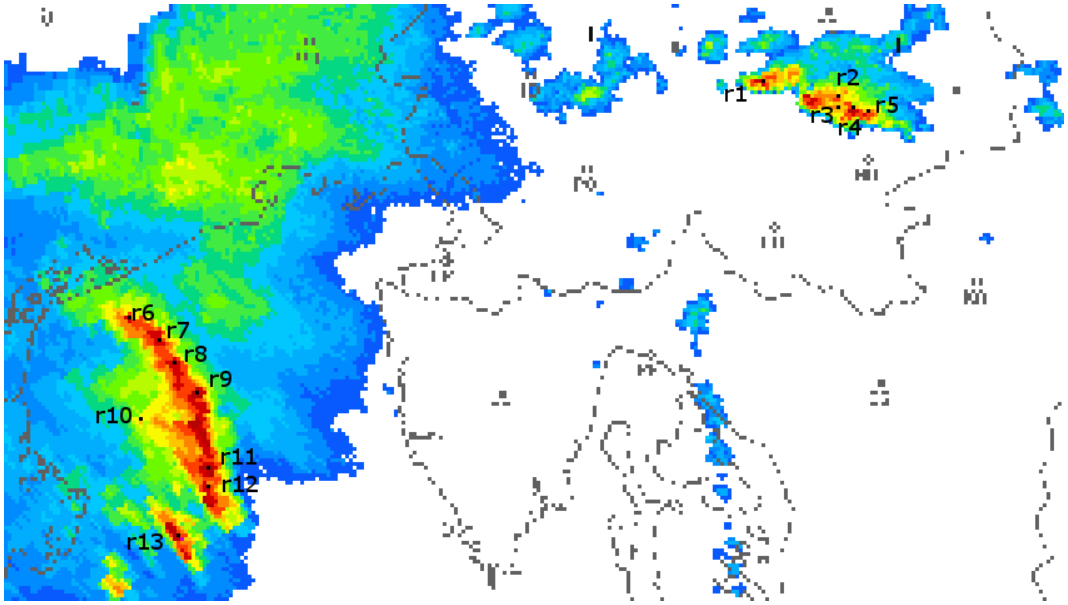

Figure 6.14: Example 2. Weather radar image taken at time $t + 50$

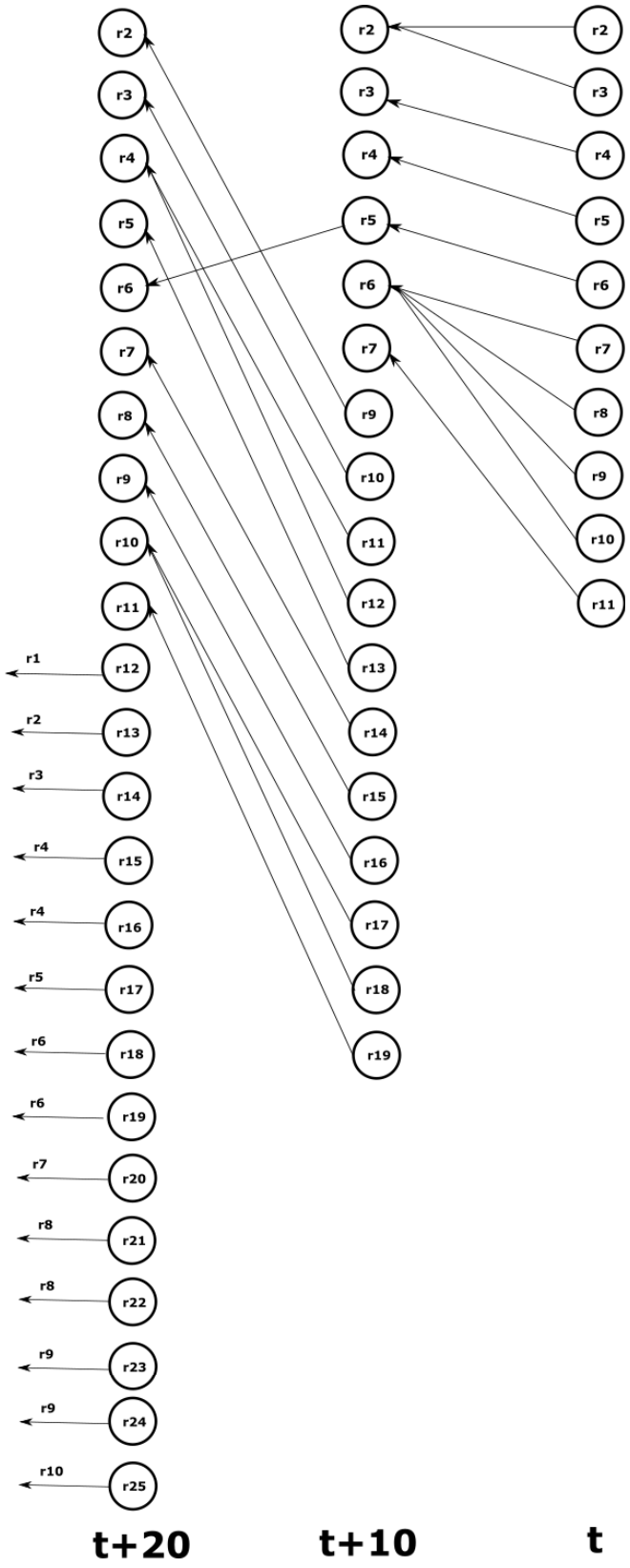

Figure 6.15: Example 2. The bifurcation diagram for the radar images taken at time t, $t + 10$, $t + 20$

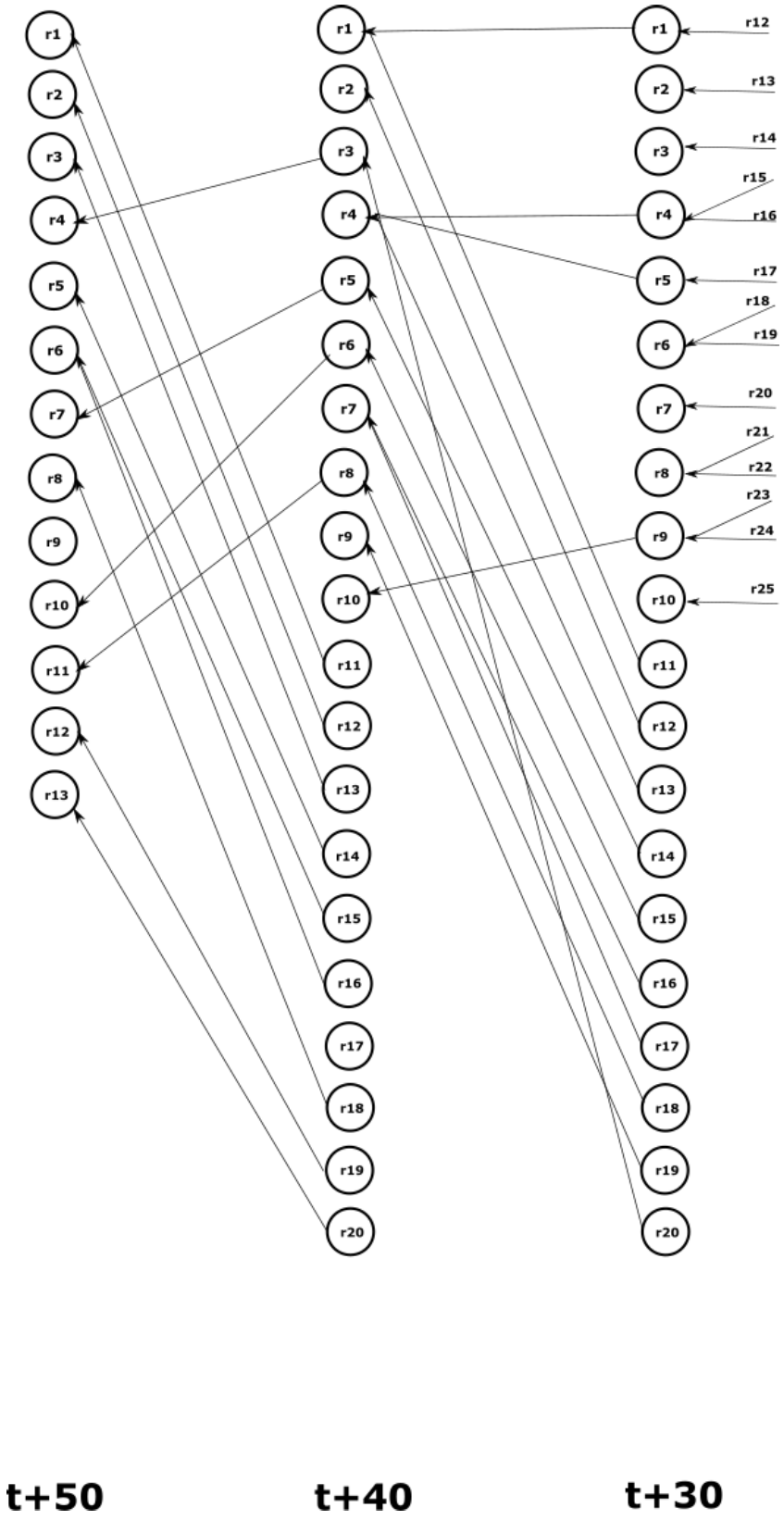

Figure 6.16: Example 2. The bifurcation diagram for the radar images taken at time $t + 30$, $t + 40$, $t + 50$